3. Sigurnosni mehanizmi operacijskog sustava

Jedanaesto poglavlje u udžbeniku L. Budin, M. Golub, D. Jakobović, L. Jelenković, Operacijski sustavi

Sadržaj

3.1. Osnovni pojmovi

- Sigurnosne prijetnje i napadi
- Sigurnosni zahtjevi
- Primjeri napada
- Sigurnosni mehanizmi OS-a

3.2. Autentifikacija

- MAC
- Autentifikacijsko kriptiranje
- Digitalni potpis

3.3. Sigurnosni protokoli

- Razmjena ključeva
- Raspodjela ključeva
- Autentifikacijski protokoli

3.4. Kontrola pristupa

- Autorizacija
- Prijava za rad
- Autentifikacijski protokol Kerberos

3.5. Infrastruktura javnog ključa

- Dijelovi PKI
- Digitalni certifikat
- X.509 autentifikacijski protokoli

3.6. Sigurnosna stijena

3.1. Osnovni pojmovi

Sigurnosne prijetnje i napadi

Sigurnosni zahtjevi

Primjeri napada

Sigurnosni mehanizmi OS-a

Ponavljanje: Sigurnosni zahtjevi

Osnovni sigurnosni zahtjevi

1. tajnost
2. autentičnost
3. neporecivost
4. integritet

Dodatni sigurnosni zahtjevi

5. kontrola pristupa
6. raspoloživost

Ponavljanje: Sigurnosni zahtjevi

Osnovni sigurnosni zahtjevi

povjerljivost

štiti informacije od neautoriziranog pristupa

1. tajnost
2. autentičnost
3. neporecivost
4. integritet

Dodatni sigurnosni zahtjevi

5. kontrola pristupa
6. raspoloživost

Ponavljanje: Sigurnosni zahtjevi

povjerljivost
Confidentiality

1. tajnost
2. autentičnost

Integrity

4. integritet

Availability

5. kontrola pristupa
6. raspoloživost

Ponavljanje: Sigurnost sustava

Sustav je siguran kada se njegovi resursi koriste i pristupa im se na za to predviđen način u svim okolnostima.

- nedostižno

Primjeri napada na sigurnost sustava

- neautorizirano čitanje podataka – narušena povjerljivost
- neautorizirana modifikacija podataka – narušen integritet
- neautorizirano brisanje podataka – narušena raspoloživost
- napad uskračivanjem usluge – narušena raspoloživost
- krađa usluge – narušeni povjerljivost i integritet

Ponavljanje: Osnovni kriptografski pojmovi

Kriptiranje vs. šifriranje

- Kerckhoffov princip: Kriptosustav mora biti siguran i onda kada su sve informacije o kriptosustavu javno poznate, osim tajnog ključa.

Podjela kriptografskih algoritama

- simetrični algoritmi
  - AES, DES, 3DES, IDEA
- asimetrični algoritmi
  - RSA, ECC, ElGamal
- funkcije za izračunavanje sažetka (*hash*)
  - SHA-1, SHA-2, SHA-3

ranjivosti i napadi	razine	zaštita
logičke greške, propusti u dizajnu, umetanje koda	programi	pokretanje u zaštićenom okruženju
nepravilno podešene postavke sustava, propusti u sustavu	OS	redovito ažuriranje sustava, rekonfiguracija postavki sustava
prisluškivanje, lažno predstavljanje	mreža	sigurnosni štit (*firewall*) kriptiranje, autentifikacija
pristup računalu, napadi na sklopovlje, napadi koji koriste sporedna svojstva	fizička razina	kriptiranje na nivou uređaja, zaštitari, zaštićene prostorije

Sigurno programiranje

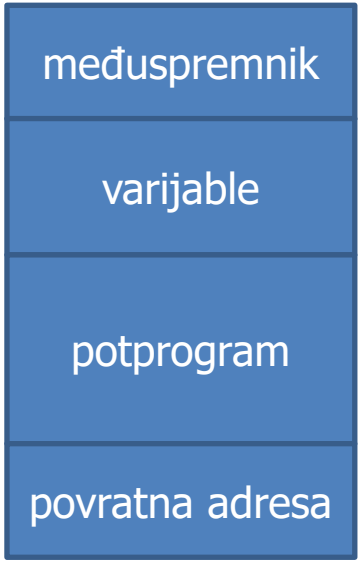
- veliki izazov
- primjer programa koji ima problema s preljevom međuspremnika (*buffer-overflow*):

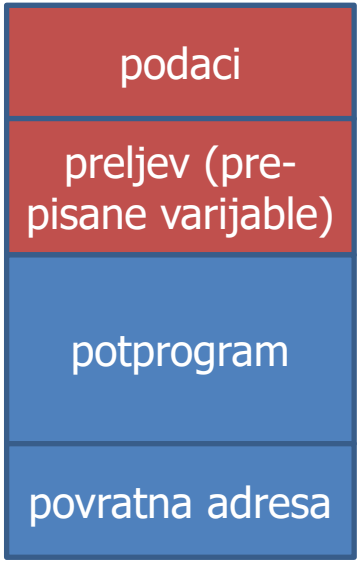
```
#include <stdio.h>
#define BUFFER SIZE 256
int main(int argc, char *argv[]){
   char buffer[BUFFER SIZE];
   if (argc < 2)
        return -1;
   else {
        strcpy(buffer,argv[1]);
        return 0;
   }
}
```

- pomaže kada programeri međusobno pregledavaju programe (*code review*) tražeći logičke pogreške

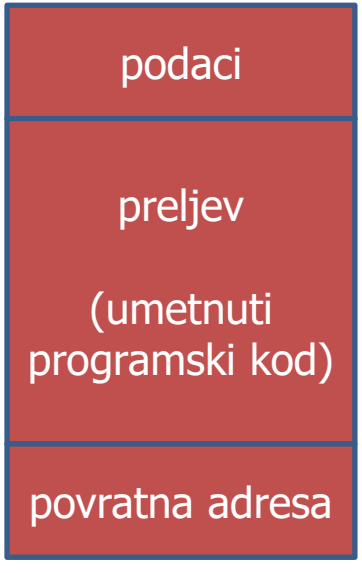
Moguće posljedice napada umetanjem koda

- engl. code injection



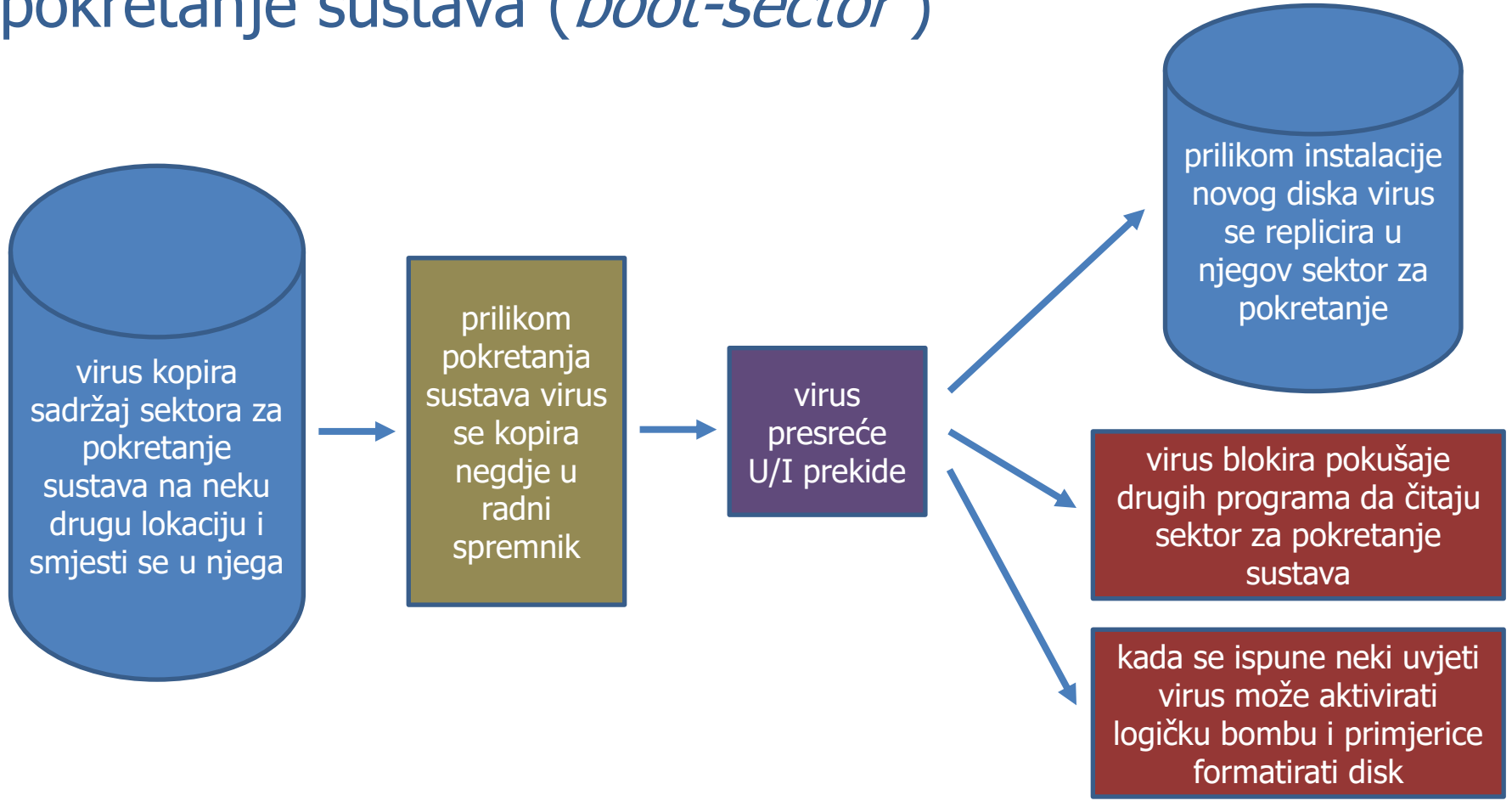


zamjena sadržaja varijabli



povratna adresa je adresa zloćudnog programa

Primjer računalnog virusa smještenog u sektor za pokretanje sustava (*boot-sector*)



Sigurnosni mehanizmi OS-a

- Kontrola pristupa
  - fundamentalni sigurnosni mehanizam OS-a
  - čemu sve pojedini proces smije pristupiti
  - upravljanje pravima pristupa
- Autentifikacija
  - prilikom prijave
  - za svaki proces zna se čiji je
- Vođenje dnevnika (engl. *logging*)
  - nadzor, otkrivanje propusta, forenzika, oporavak od pogrešaka
- Kriptiranje datotečnog podsustava
- Ažuriranje operacijskog sustava

3.2. Autentifikacija poruka

MAC

Autentifikacijsko kriptiranje

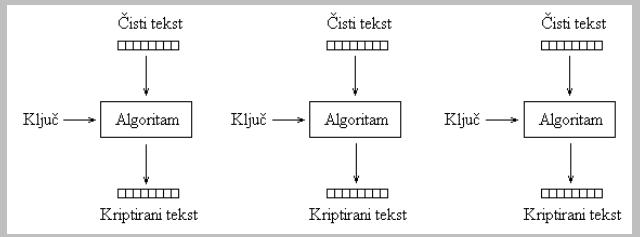
Digitalni potpis

Autentifikacija

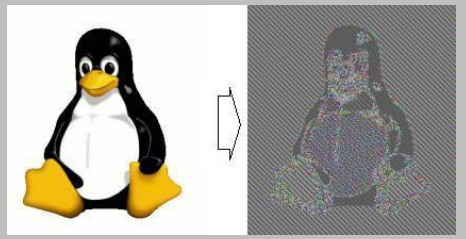
- korisnika
  - Prijava za rad
  - time ćemo se baviti naknadno
- poruka
  - načinima kriptiranja MAC (*Message Authentication Code*) i HMAC
  - hibridnim postupcima EtM, E&M, MtE te načinom kriptiranja GCM ako želimo uz autentičnost osigurati i tajnost
  - autentifikacijsko kriptiranje osigurava tajnost i autentičnost
  - digitalni potpis osigurava autentičnost, integritet i neporecivost
    - digitalni pečat osigurava i tajnost
    - autentičnost se u PKI sustavu utvrđuje certifikatom

Načini kriptiranja

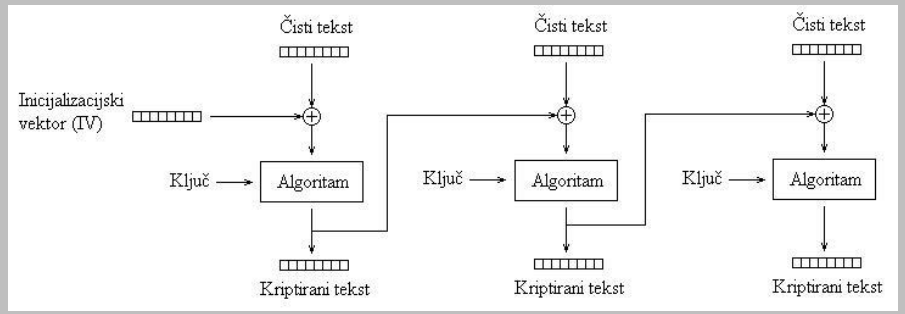
ECB



cilj je onemogućiti da isti blokovi jasnog teksta daju isti blok kriptiranog teksta ...

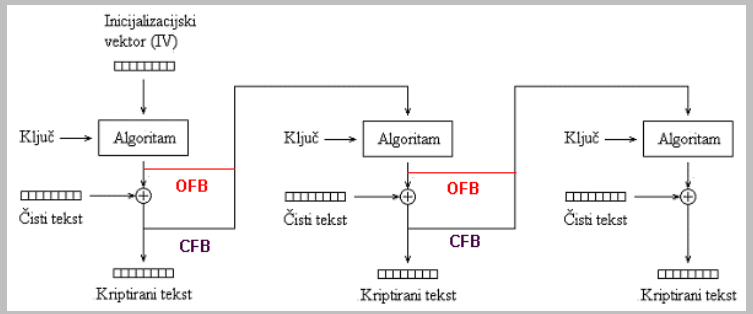


CBC

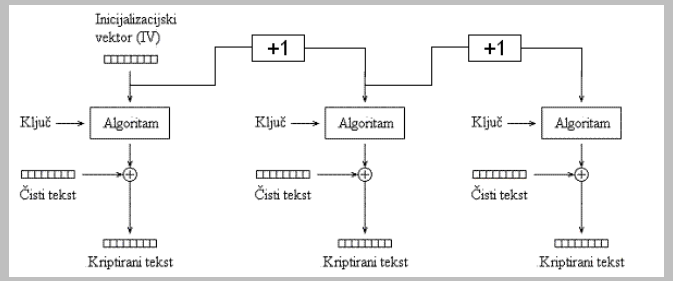


... te da se blok simetrični algoritmi mogu koristiti za kriptiranje toka podataka

CFB i OFB

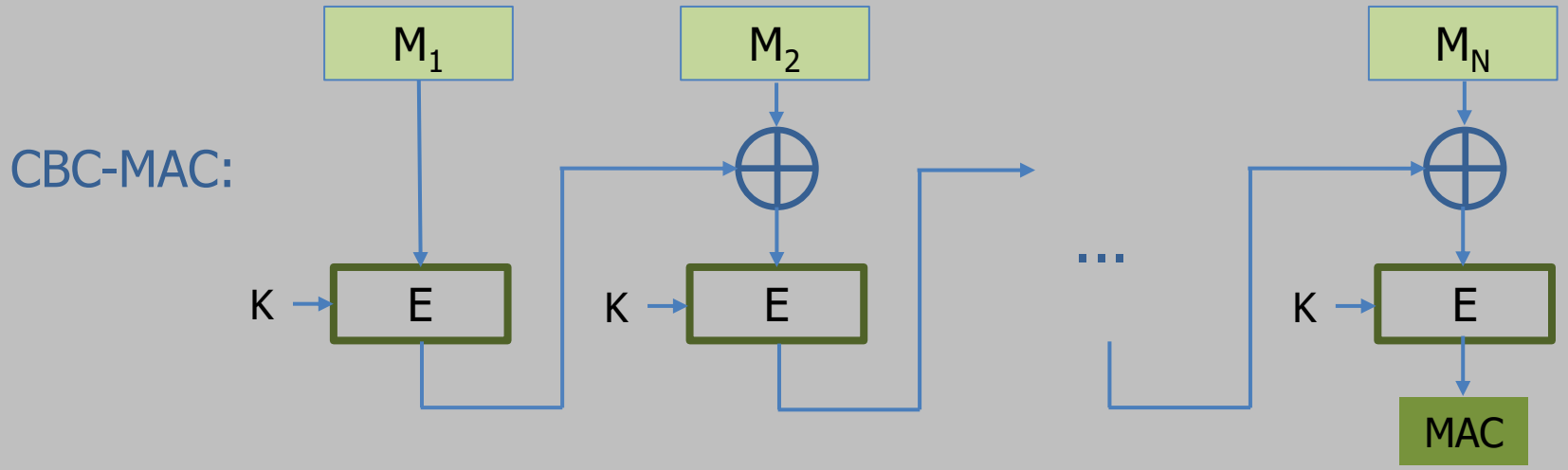


CTR

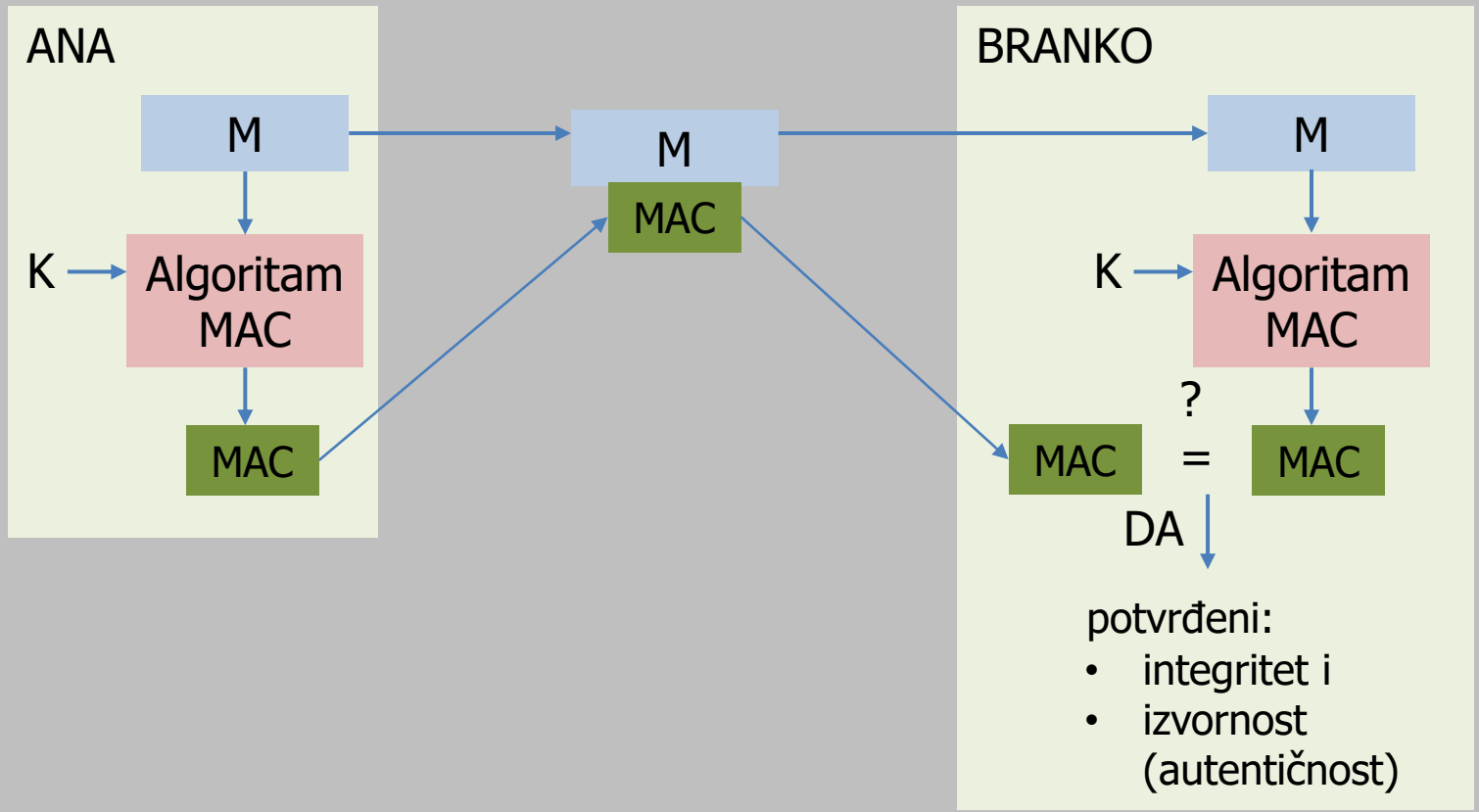


Način kriptiranja koji osigurava autentičnost i integritet MAC (*Message Authentication Code*)

- autentičnost pošiljatelja osigurava se simetričnim kriptiranjem
  - poruku je zaštitio netko od onih koji imaju tajni ključ K



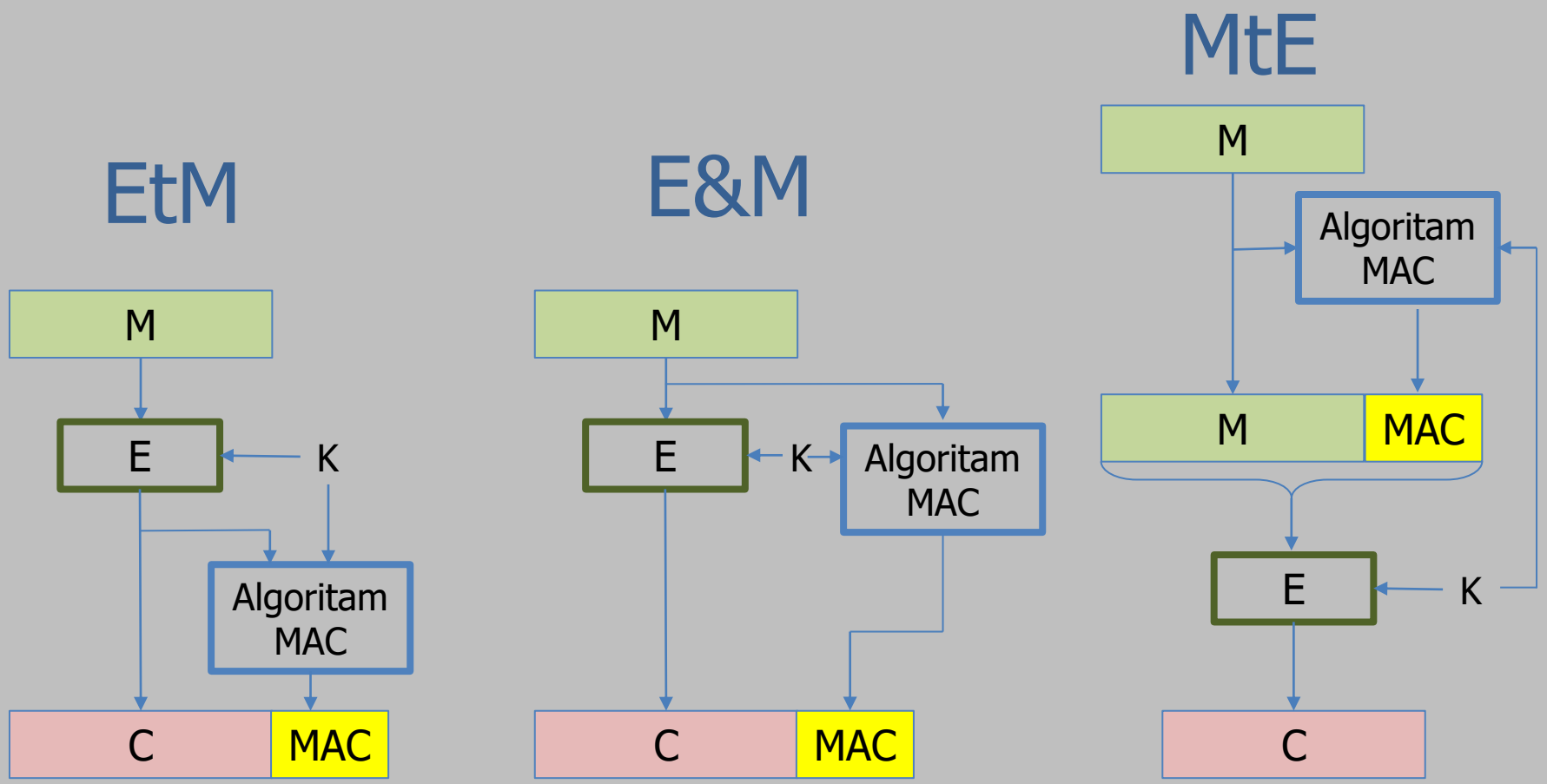
Primjer kako se može koristiti dodatak poruci MAC



Način kriptiranja koji osigurava autentičnost HMAC

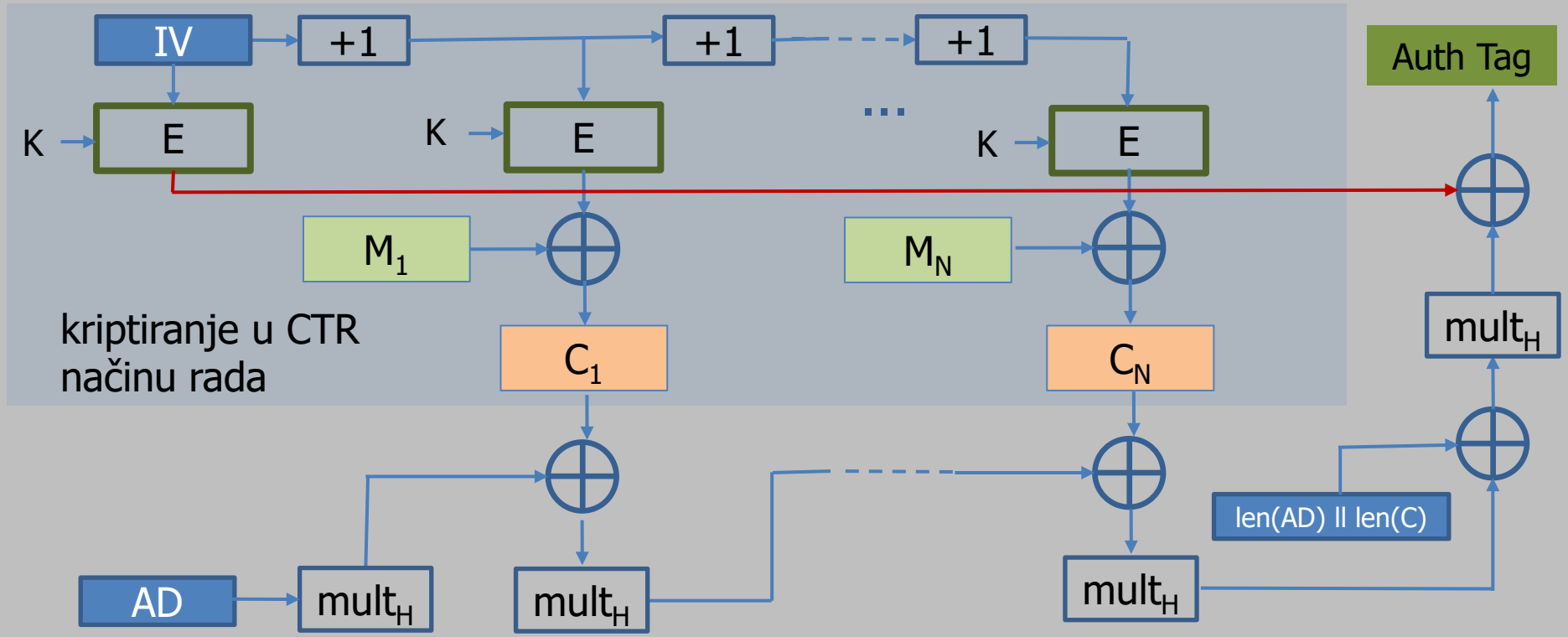
- umjesto blok simetričnog algoritma koristi funkciju za izračunavanje sažetka poruke
- *keyed-Hash Message Authentication Code*
  - HMAC_MD5
  - HMAC_SHA1
  - HMAC_SHA256
  - HMAC_SHA3
- $HMAC(K,M) = H\{(K' \oplus opad) \,||\, H[(K' \oplus ipad) \,||\, M]\}$
  - K' = H(K) ako je K veći od veličine bloka, inače K' = K
  - konstanta *opad (outer padding*) = 0x5c5c5c...5c5c
  - konstanta *ipad* (*inner padding*) = 0x363636...3636
  - *opad* i *ipad* su veličine jednog bloka

Kako uz integritet i autentičnost osigurati i tajnost?



GCM - *Galois/Counter Mode*

- način autentifikacijskog kriptiranja koji osigurava autentičnost i tajnost, a primjenjiv je samo za simetrične blok algoritme s veličninom bloka 128 bita
- varijanta *Galois Message Authentication Code, GMAC* – samo za autentifikaciju



Autentifikacijska kriptografija

- Natječaj CAESAR (*Competition for Authenticated Encryption: Security, Applicability, and Robustness*)
  - nedostatak klasičnih autentifikacijskih kriptografskih shema poput *EtM, E&M i MtE* je upravo u primjeni više algoritama
  - završio 20.3.2019. objavljeno 3 pobjednika u 3 kategorije i 5 rezervna algoritma

- NIST-ov natječaj za novi algoritam prilagođen okruženju s ograničenim računalnim resursima (*lightweigt cryptography*)
  - algoritam treba osim simetričnog uključivati i autentifikacijsko kriptiranje (*Authenticated Encryption with Associated Data*, AEAD)
  - natječaj je u tijeku

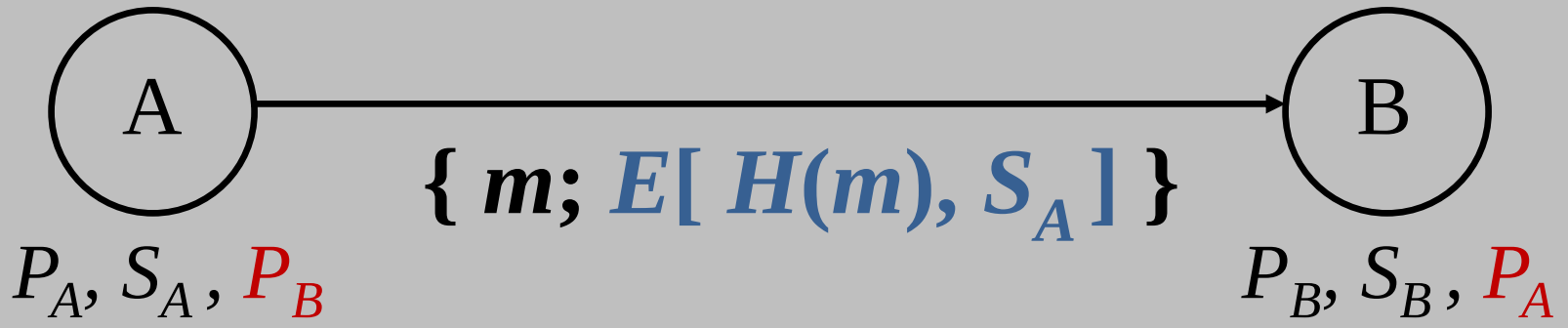
Pobjednici na natječaju CAESAR

Pobjednici su birani u tri kategorije:

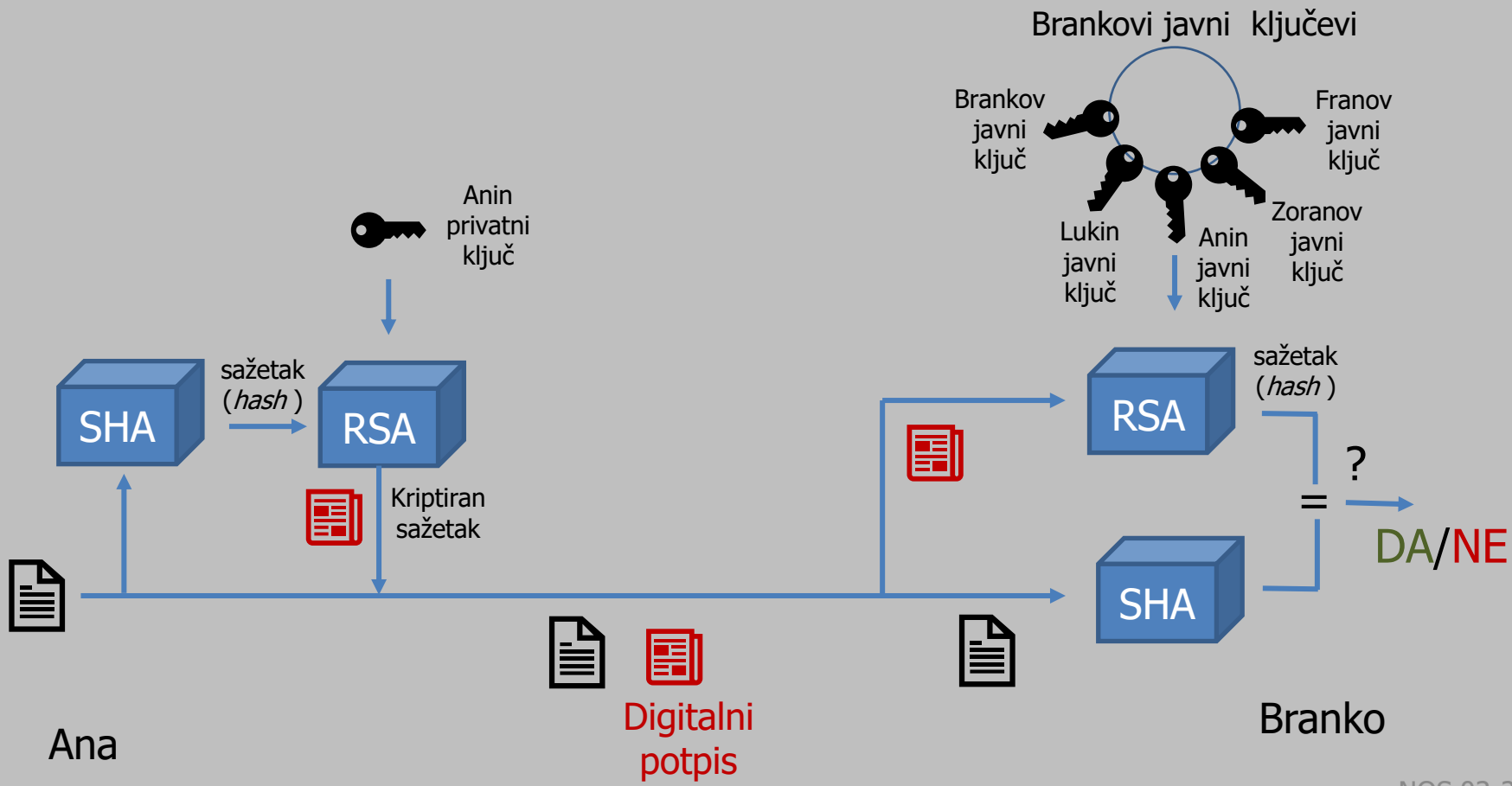
1. Algoritmi koji su najmanje zahtjevni na računalne resurse (*Lightweight applications - resource constrained environments*)
   - prvi izbor: Ascon (web)
   - drugi izbor: ACORN
2. Algoritmi visokih performansi (*High-performance applications*)
   - prvi izbor: AEGIS-128
   - drugi izbor: OCB
3. Sigurnost (*Defense in depth*)
   - prvi izbor: Deoxys-II
   - drugi izbor: COLM ili AES-COPA ili ELmD

Digitalni potpis

- dodatak poruci koji služi za
  - utvrđivanje besprijekornosti informacije (integritet i neporecivost) i za
  - identifikaciju pošiljatelja (autentičnost)
- ne osigurava tajnost!

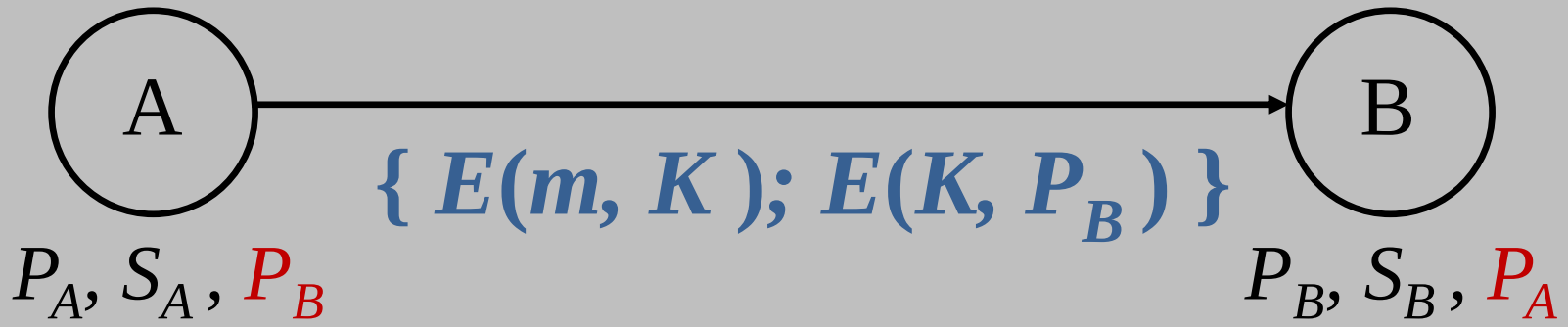


Digitalni potpis
Postupak potpisivanja i provjere



Digitalna omotnica

- osigurava tajnost
- pošiljatelj kriptira poruku proizvoljnim ključem K simetričnim algoritmom kriptiranja
- simetrični (sjednički ili tajni) ključ K se kriptira javnim ključem primatelja P_B
- kriptirana poruka i kriptirani ključ čine digitalnu omotnicu



Digitalni pečat (1/2)

- digitalni pečat osigurava četiri sigurnosna zahtijeva:
  - tajnost
  - autentičnost
  - integritet i
  - neporecivost
- digitalni pečat je digitalno potpisana digitalna omotnica

$$\{ E(m,K); E(K,P_B) \}; \; E\{ H[E(m,K); E(K,P_B)], S_A \}$$

Digitalni pečat (2/2)

- češće se koristi obrnuti postupak:
  1. digitalno se potpiše poruka
  2. poruka s potpisom se kriptira slučajno generiranim tajnim ključem *K*
  3. na kraju se dodaje kriptirani ključ javnim ključem primatelja
- digitalni pečat je digitalna omotnica s potpisanom porukom:

$$E\{\,[\,m;\ \underbrace{E(H(m), S_A)}_{\text{digitalni potpis}}\,],\ K\,\};\ \ E(K, P_B)$$

Autentifikacija poruka

algoritam	autentičnost	tajnost	neporecivost	integritet
MAC HMAC	+*			+
EtM, E&M, MtE GCM	+*	+		+
autentifikacijsko kriptiranje	+*	+		+
digitalna omotnica		+		
digitalni potpis	+		+	+
digitalni pečat	+	+	+	+

* ne autentificira pošiljatelja već jamči da je poruku poslao „onaj koji ima tajni ključ”

3.3. Sigurnosni protokoli

Razmjena ključeva

Raspodjela ključeva

Autentifikacijski protokoli

Sigurnosni protokoli

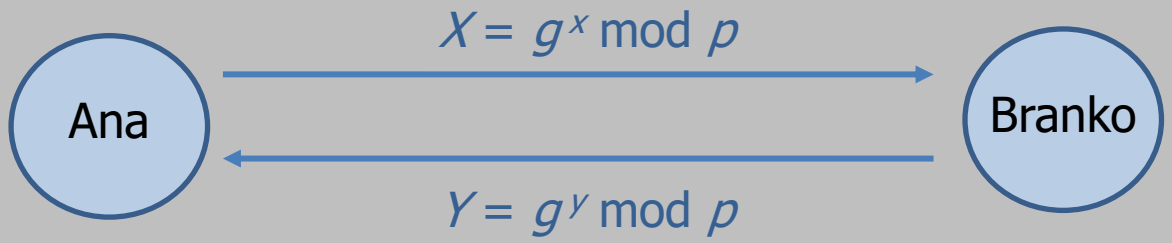
Diffie - Hellmanov postupak

- služi za razmjenu tajnog ključa
- Ana i Branko se unaprijed slože o dva vrlo velika broja n i g:
- $$\text{nzd}(g, n) = 1$$
- najpraktičnije: za n odabrati veliki prosti broj p
- g i p se mogu javno objaviti
- Ana odabire veliki nasumični prirodni broj x i šalje Branku:

$$X = g^x \bmod p$$

- Branko odabire veliki nasumični prirodni broj y i šalje Ani:

$$Y = g^y \bmod p$$



- Ana dobiva Y i izračuna zajednički ključ:

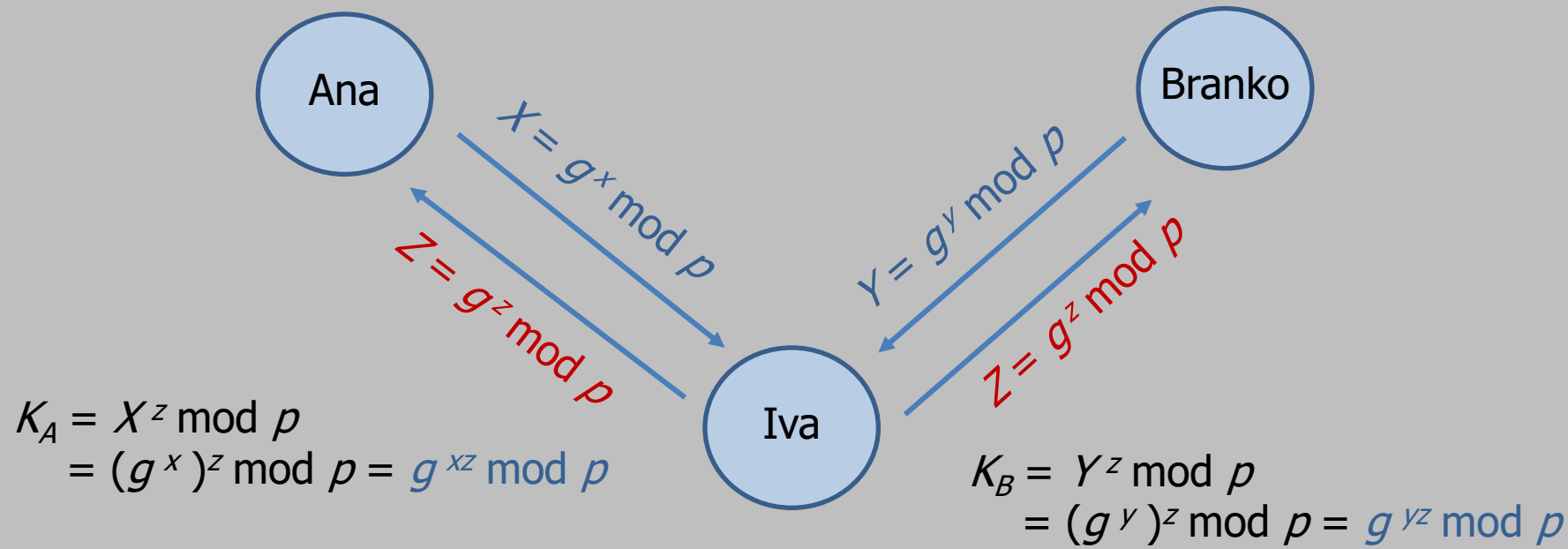
$$K = Y^x \bmod p = (g^y)^x \bmod p = g^{xy} \bmod p$$

- Branko također:

$$K = X^y \bmod p = (g^x)^y \bmod p = g^{xy} \bmod p$$

Napad *čovjek u sredini* (engl. *man in the middle*)

- napadač komunicira s Anom i Brankom (lažno se predstavljajući) uz pomoć dva ključa K_A i K_B



Raspodjela ključeva u zatvorenom simetričnom kriptosustavu

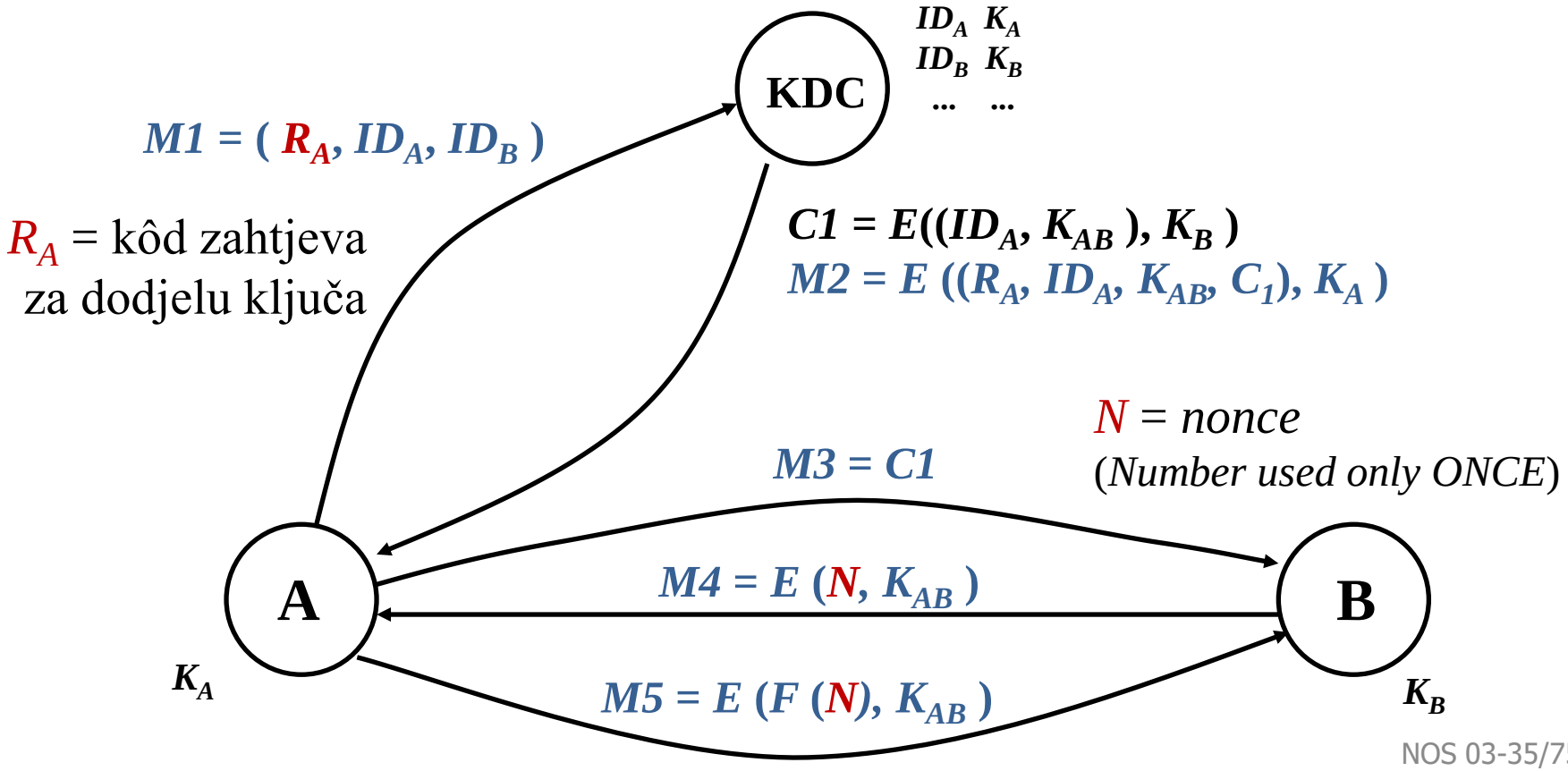
Raspodjela ključeva prema Needhamu i Schroederu

- za N sudionika: ukupno $N(N-1)/2$ tajnih ključeva i svaki sudionik bi morao pohraniti $N-1$ ključeva $\Rightarrow$ ozbiljno je ugrožena sigurnost!
- rješenje: pouzdani poslužitelj u kojem imaju svi povjerenje

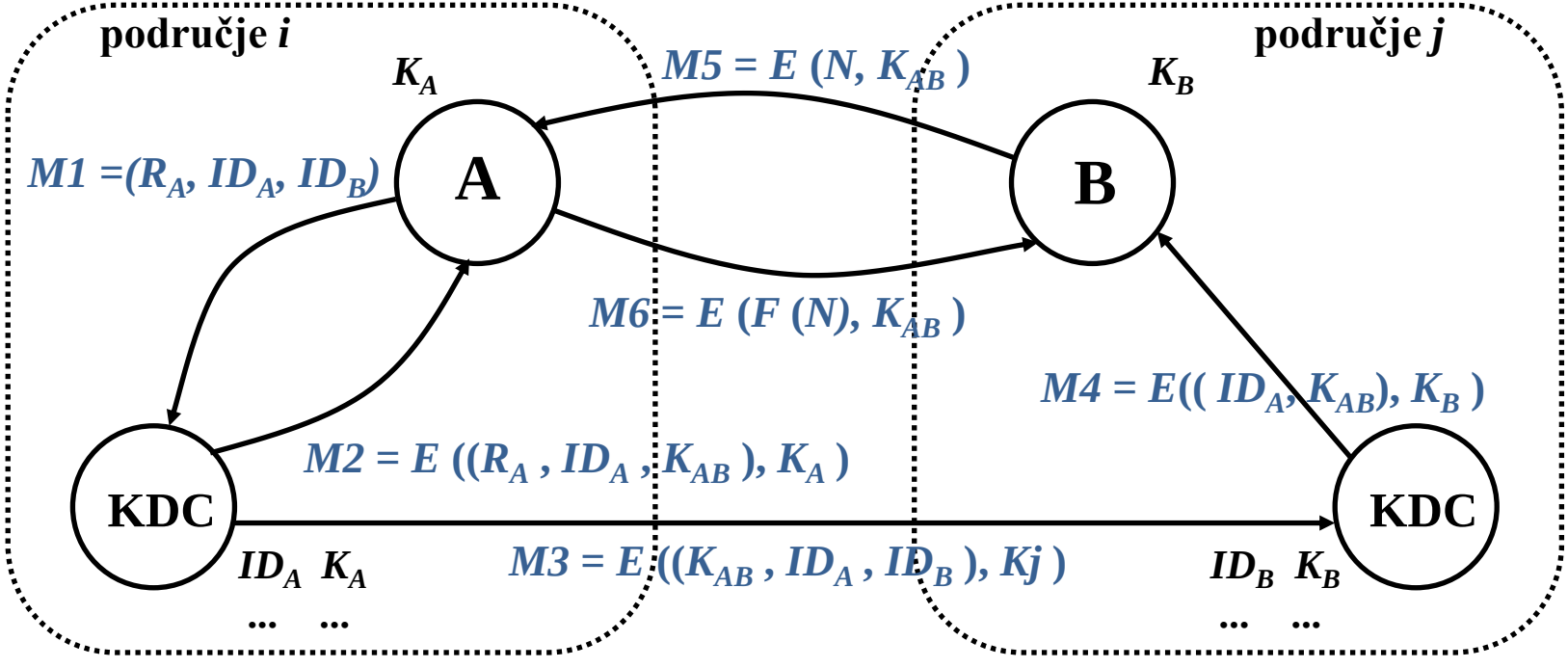
Centar za raspodjelu ključeva (*Key Distribution Center - KDC*)

- potencijalni sudionici moraju se unaprijed prijaviti
- dodjeljuje im se tajni ključ za komuniciranje s KDC
- KDC obznanjuje identifikatore svih prijavljenih sudionika a zadržava u tajnosti pripadnu tablicu tajnih ključeva

Raspodjela ključeva u zatvorenom simetričnom kriptosustavu



Raspodijeljena raspodjela ključeva u zatvorenom simetričnom kriptosustavu



Primjer 11.9. - u zatvorenom sustavu je 100 sudionika

a) Bez centra treba unaprijed podijeliti 100 x 99 / 2 = 4950 ključeva. Svaki sudionik mora čuvati 99 ključeva.

b) S jednim *KDC* treba unaprijed podijeliti 100 ključeva.
   - Svaki sudionik čuva samo svoj (1) ključ, a KDC 100 ključeva.
   - Nedostatak : smanjena pouzdanost i preopterećenost

c) U raspodijeljenom sustavu postoji 10 x 9 / 2 = 45 ključeva za komunikaciju između centara. Svaki KDC čuva njih 9.
   - U svakom područnom KDC postoji 10 ključeva za komuniciranje sa sudionicima unutar područja.
   - Svaki sudionik čuva samo svoj (1) ključ za komuniciranje s područnim centrom.
   - Područni KDC mora čuvati i tih 10 ključeva, tako da on čuva ukupno 9 + 10 = 19 ključeva.
   - U tom se sustavu koristi se ukupno

     10 x 10 + 45 = 145 ključeva.

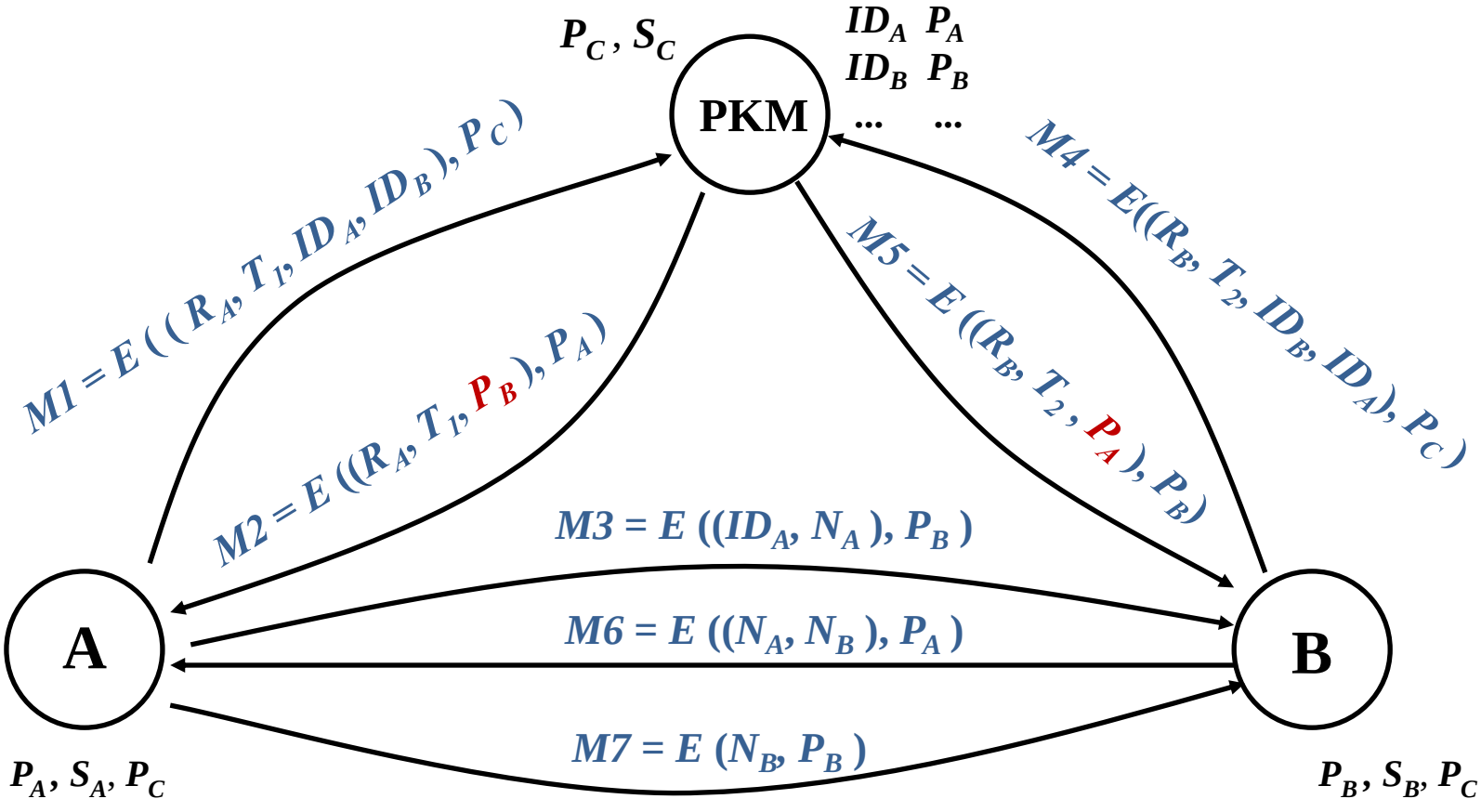
Raspodjela ključeva u zatvorenom asimetričnom kriptosustavu

- raspodjeljuju se samo javni ključevi
- problem: svatko se može lažno predstaviti

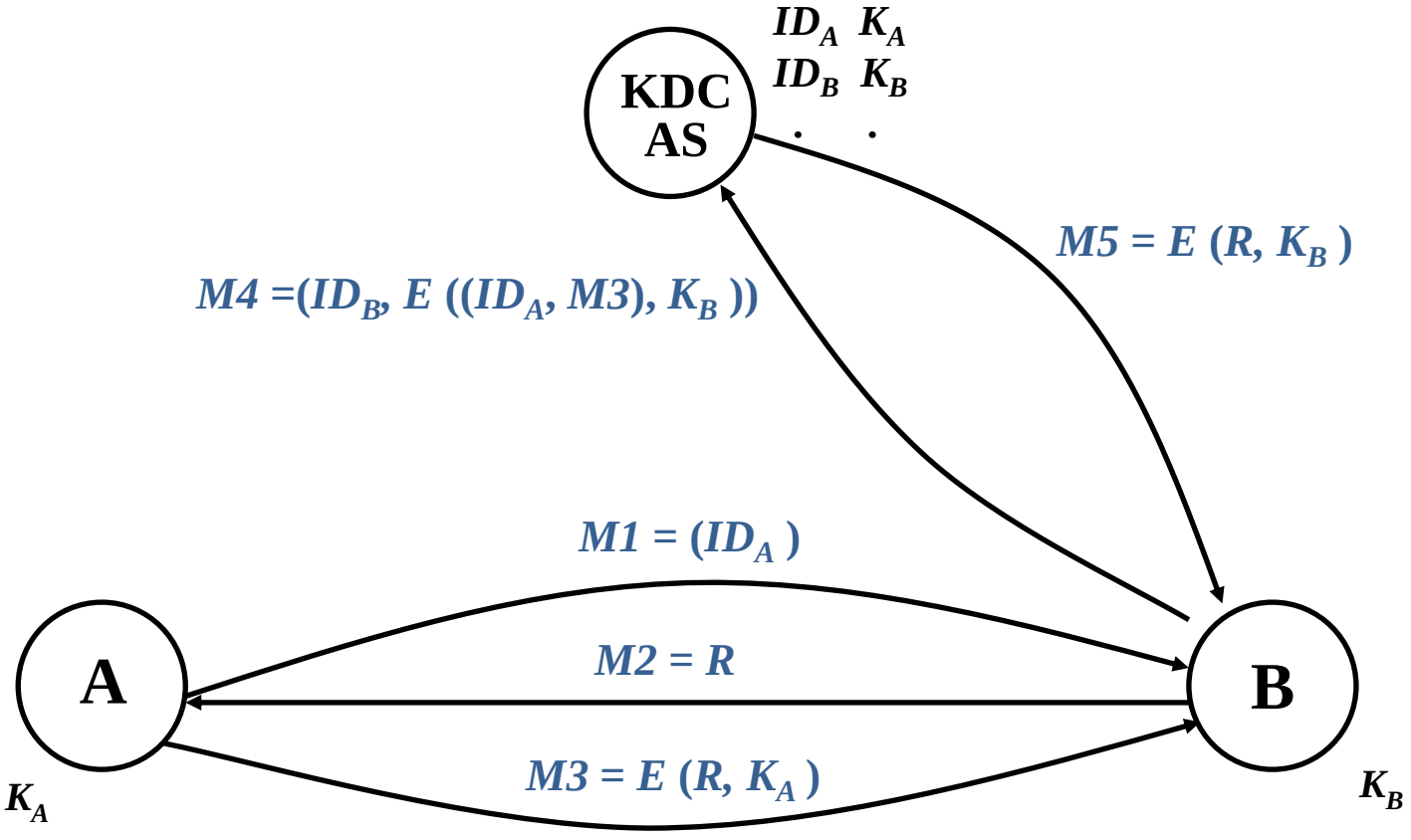
Centar za raspodjelu ključeva (*Public Key Manager – PKM*)

- potencijalni sudionici moraju se unaprijed prijaviti i autentificirati
- privatni ključ sudionici čuvaju za sebe
- PKM obznanjuje identifikatore svih prijavljenih sudionika i čuva pripadnu tablicu javnih ključeva
- prije raspodjele ključeva potrebno je obaviti autentifikaciju

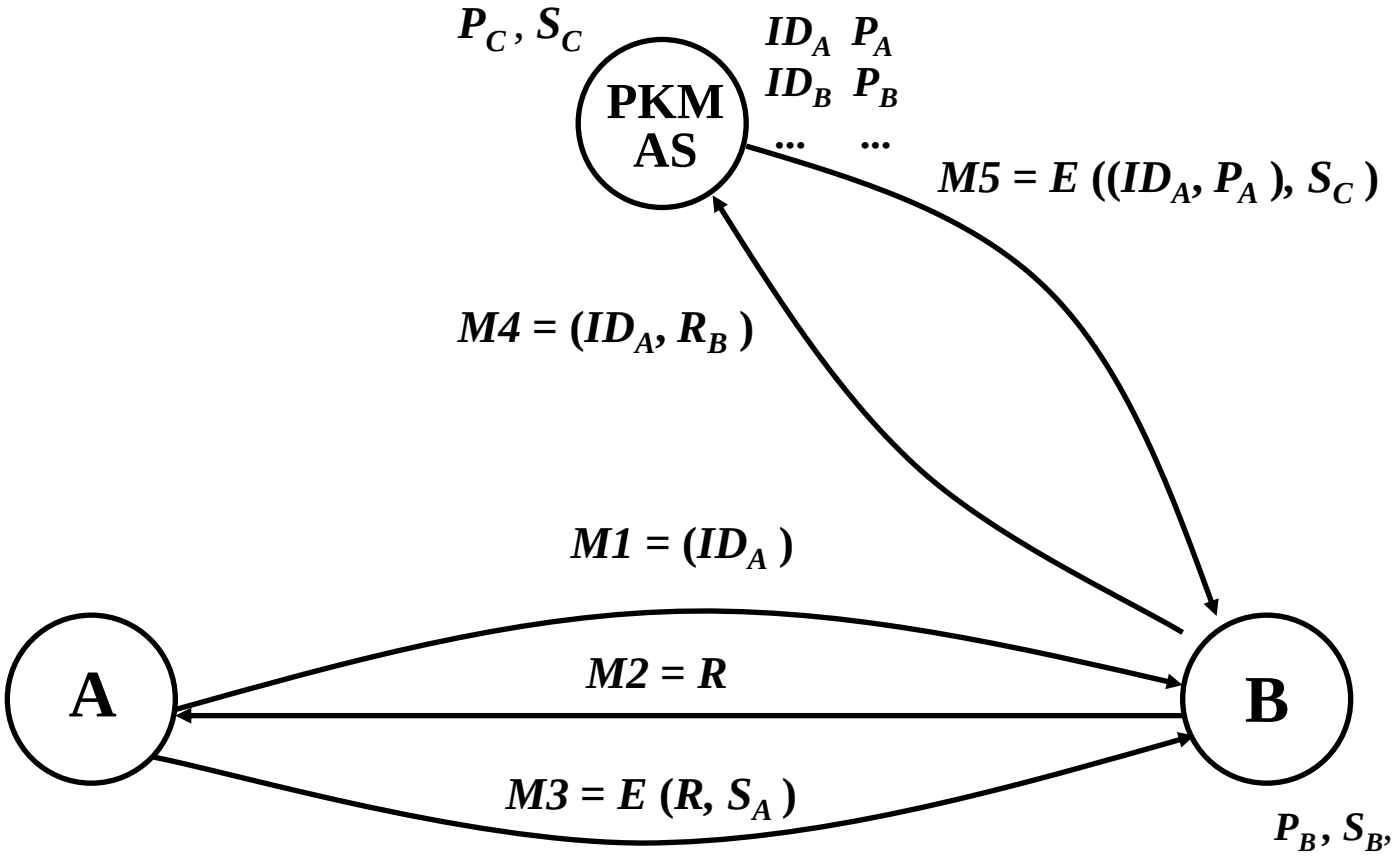
Raspodjela ključeva u zatvorenom asimetričnom kriptosustavu



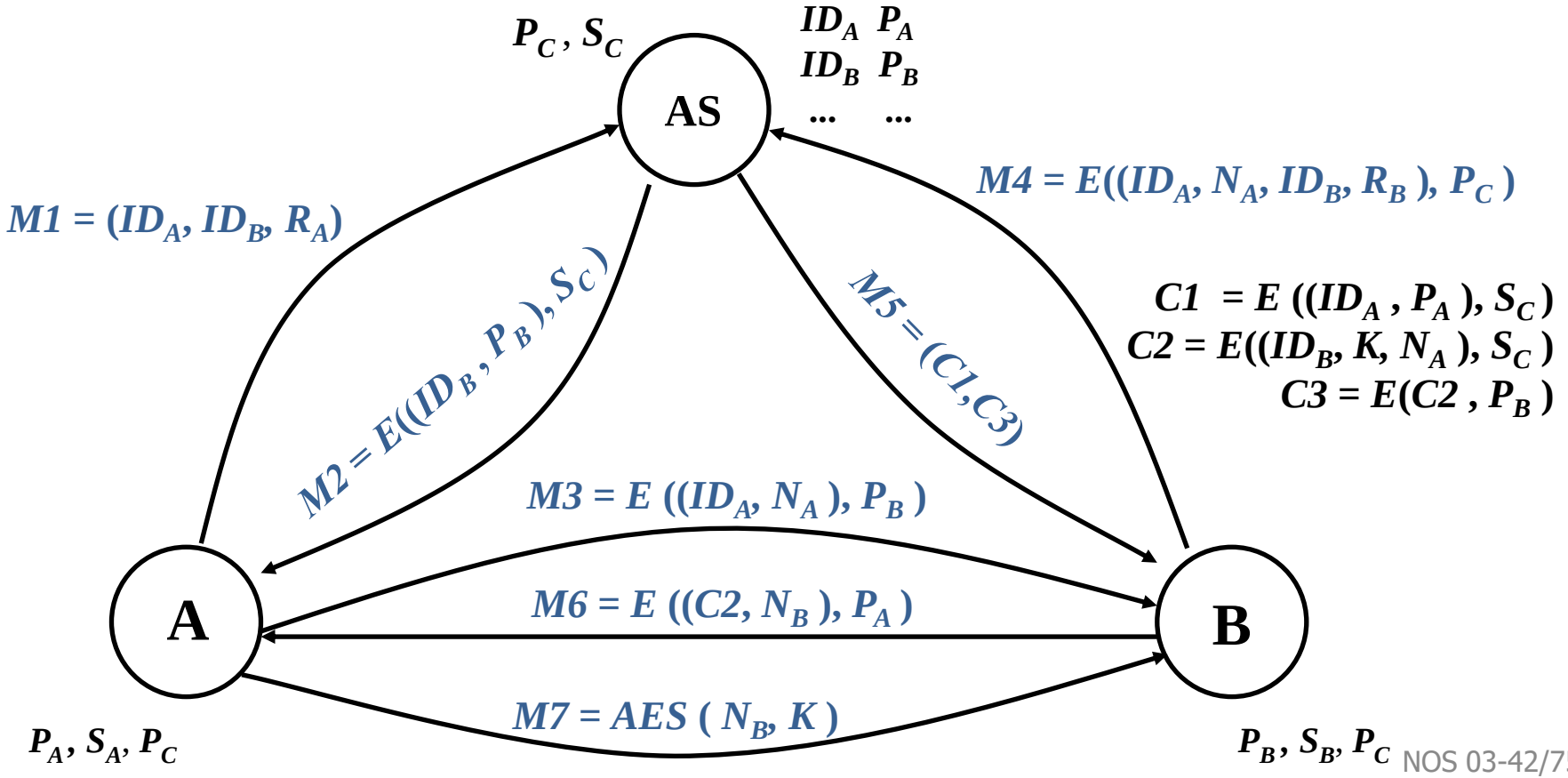
Jednostrana autentifikacija u zatvorenom simetričnom kriptosustavu



Jednostrana autentifikacija u zatvorenom asimetričnom kriptosustavu



Oboostrana autentifikacija u zatvorenom asimetričnom kriptosustavu



3.4. Kontrola pristupa

Autorizacija

Prijava za rad

Autentifikacijski protokol Kerberos

Kontrola pristupa

- najvažniji sigurnosni mehanizam operacijskog sustava

Tko može čemu pristupiti i na koji način.

- „tko” su subjekti
  - korisnici i procesi
- „čemu” su objekti
  - datoteke, mrežne pristupne točke, procesi
  - mehanizmi operacijskog sustava
    - redovi poruka
    - semafori
    - dijelovi spremnika (zajednički dijelovi spremnika)
- „način” pristupa ili dozvole
  - npr. rwx
  - ovisi o vrsti objekta

Kontrola pristupa se obavlja autorizacijom

- zaštita pristupanja pojedinim sredstvima = autorizacija
  - autentifikacija + provjera prava pristupa (*access control*)
  - mehanizmi *dopuštanja pristupa* (engl. *access control*) sredstvima nazivaju se *autorizacijom pristupa* (engl. *authorization*)
- **subjekti**: korisnici ili njihovi procesi ili čak neke dretve unutar tih procesa
- **objekti** zaštite: sredstva koja se zaštićuju
- zaštitna pravila (engl. *protection rules*)
  - za svaki par subjekt-objekt treba odrediti pravo pristupa
  - obuhvaća i način na koji se objekt smije upotrebljavati
  - r,w,x ili prazno polje = nema prava pristupa
  - mogu prikazati u obliku **matrice pristupa** (engl. *access matrix*)
  - svaki subjekt dobiva svoj redak i svaki objekt svoj stupac

OBJEKTI

SUBJEKTI

Č, P	I		Č	
		P		P
Č				
	I			
		Č		

Alternativni način zapisa - liste :

- **Lista prava pristupa objektu** (engl. *access control list*)
  - neprazni elementi stupaca matrice pristupa
- **Lista dozvola za pristup objektima** (engl. *capability tickets*)
  - neprazni elementi pojedinih redova matrice

Prijava za rad

- ime korisnika i lozinka (*user name*, *password*)
- iz imena se izvodi identifikator korisnika (*user identifier*)
- slabosti:
  - korisnici mogu sami lako otkriti svoje podatke jer ih obično zapisuju
  - napadači su vrlo domišljati pri otkrivanju identifikatora i lozinki
  - datoteke s identifikatorima i lozinkama su meta napadača
  - engleski rječnik ima približno 350 000 riječi
    - napad "grubom silom" traje u najgorem slučaju desetak sekundi

Metode za povećanje sigurnosti prilikom prijave za rad (1/2)

Lozinke

- broj mogućih lozinki mora biti velik čime se smanjuje vjerojatnost pogađanja lozinke
  - po mogućnosti treba sadržavati mala i velika slova, brojeve i interpunkcijske znakove
- OS prilikom registracije treba
  - onemogućiti jednostavnu i/ili kratku lozinku ili
  - predložiti slučajno generiranu
- postupak prijave mora biti takav da se dozvoljava samo ograničeni broj ponavljanja netočne lozinke
- operacijski sustav mora pohranjivati pokušaje neovlaštenog pristupa kako bi se olakšala naknadna istraga

Metode za povećanje sigurnosti prilikom prijave za rad (2/2)

Lozinke (nastavak)

- umjesto $C = E(P, K)$ koristiti $C = E(P, P)$ ili funkciju za izračunavanje sažetka (*hash*)
- nadopuniti lozinku nasumičnim brojem (engl. *salt*) koji se čuva u posebnoj tablici i mijenja se kod svake promjene lozinke kako dvije iste lozinke ne bi imale jednaki kriptirani tekst Lozinke koje se koriste samo jednom (engl. *One-time passwords*)

Biometrijske metode

Višerazinska autentifikacija (engl. *Multi-factor authentication*)

- npr. uz lozinku se koristi USB "dongle" ili neka biometrijska metoda

Primjer čuvanja lozinki u operacijskim sustavima Linux i Unix

- lozinke se čuvaju u datoteci `/etc/shadow`
- po jedan zapis za svakog korisnika ima oblik:

```
korisnik:$alg$kriptirana_lozinka:pr:min:max:upoz:neak:ist
```

- **korisnik** = korisničko ime (*username*)
- **alg** = algoritam kriptiranja
  - 1 = MD5
  - 2a = Blowfish
  - 2y = Blowfish
  - 5 = SHA-256
  - 6 = SHA-512
- **pr** = dan kada je zadnji puta izmijenjena lozinka (od 1.1.1970.)
- **min** = minimalni broj dana da se može ponovno promijeniti lozinka
- **max** = maksimalni broj dana koliko može biti lozinka valjana
- **upoz** = broj dana prije nego istekne lozinka kada se korisnku izdaje upozorenje
- **neak** = broj dana nakon što lozinka istekne kada će se onemogući korisnički račun
- **ist** = dan kada korisnička lozinka ističe (od 1.1.1970.)

Primjer čuvanja lozinki u operacijskim sustavima Linux i Unix

- informacije o korisnicima se čuvaju u datoteci `/etc/passwd`
- po jedan zapis za svakog korisnika ima oblik:

`korisnik:x:UID:GID:UIDinfo:home:ljuska`

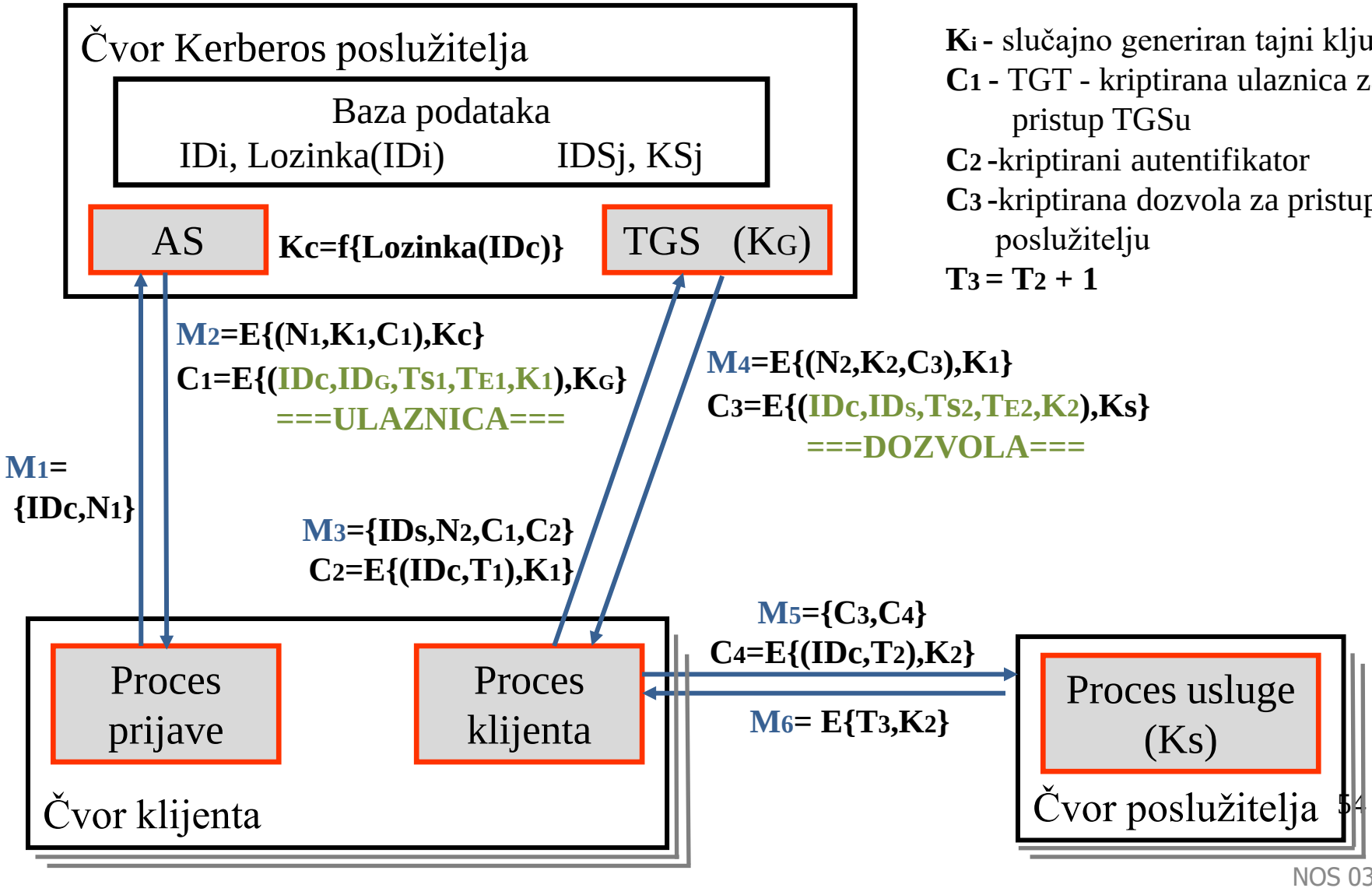
- **`korisnik`** = korisničko ime (*username*)
- **`x`** = oznaka 'x' označava da je pohranjena kriptirana lozinka u `/etc/shadow`
- **`UID`** = identifikacijski broj korisnika
- **`GID`** = identifikacijski broj grupe
- **`UIDinfo`** = polje u koje se upisuje komentar o korisniku, primjerice puno ime i prezime, telefon i sl.
- **`home`** = korisnički direktorij (*home directory*)
- **`ljuska`** = naredba ili korisnička ljuska. Uobičajeno je korisnička ljuska npr. `/bin/bash`

Autentifikacijski protokol Kerberos

- počeo se razvijati 1978. godine na Massachusets Instutute of Technology (MIT)
- pretpostavka: pouzdana računala, ali je mreža nepouzdana
- koristi simetrični kriptosustav (izvorno: DES)
- treća strana kojoj svi vjeruju
- traži unos lozinke samo jednom (*single sign-on*) i to na početku sjednice
- lozinka ne putuje mrežom
- osjetljivi podaci se prenose u kriptiranom obliku

Sustav Kerberos sastoji se od

- čvora klijenta (*client node*)
- čvora poslužitelja (*application server node*) – obavlja traženu uslugu
- čvora Kerberos poslužitelja – sastoji se od:
  - baze podataka:
    - identifikatori
    - lozinke
    - tajni ključevi svih poslužitelja u sustavu
  - poslužitelja (ili procesa) za utvrđivanje autentičnosti (*authentication server – AS*)
  - poslužitelja za dodjelu ulaznica za pristup pojedinim uslugama (*ticket granting server – TGS*)



Ki - slučajno generiran tajni ključ
C1 - TGT - kriptirana ulaznica za pristup TGSu
C2 -kriptirani autentifikator
C3 -kriptirana dozvola za pristup poslužitelju
T3 = T2 + 1

Autentifikacijski sustav Kerberos

- osigurava tri nivoa zaštite:
  - provjera autentičnosti samo na početku (mrežni datotečni sustav na MIT mreži)
  - "sigurne poruke" – uz poruku u jasnom obliku šalje se i kriptirani autentifikator
  - "privatne poruke" – kriptirana poruka i kriptirani autentifikator
- distribucije Kerberosa donose kerberizirane verzije najpopularnijih aplikacija (npr. `rlogin`, `telnet`, `ftp` ...)
- ograničenja i nedostaci:
  - svaki program treba biti "kerberiziran"
  - nema autorizacije
  - Kerberos server mora biti fizički zaštićen
  - kako sigurno pohraniti tajne ključeve?
  - podliježe strogim američkim zakonima o izvozu kriptotehnologije

3.5. Infrastruktura javnog ključa

Dijelovi PKI

Digitalni certifikat

X.509 autentifikacijski protokoli

Infrastruktura javnih ključeva
PKI – Public Key Infrastructure

- skup tehnologija, protokola, normi i usluga koji zajedno omogućuju sigurnu komunikaciju temeljenu na sustavu javnih ključeva preko nesigurnih mreža

PKI infrastruktura trebala bi pružiti sljedeće:

- integritet elektronički primljene ili poslane poruke
- sigurnost u identitet pošiljaoca i primaoca informacije
- pouzdanost vremena i datuma slanja informacije
- formalnopravnu valjanost elektroničke poruke u sudskim procesima

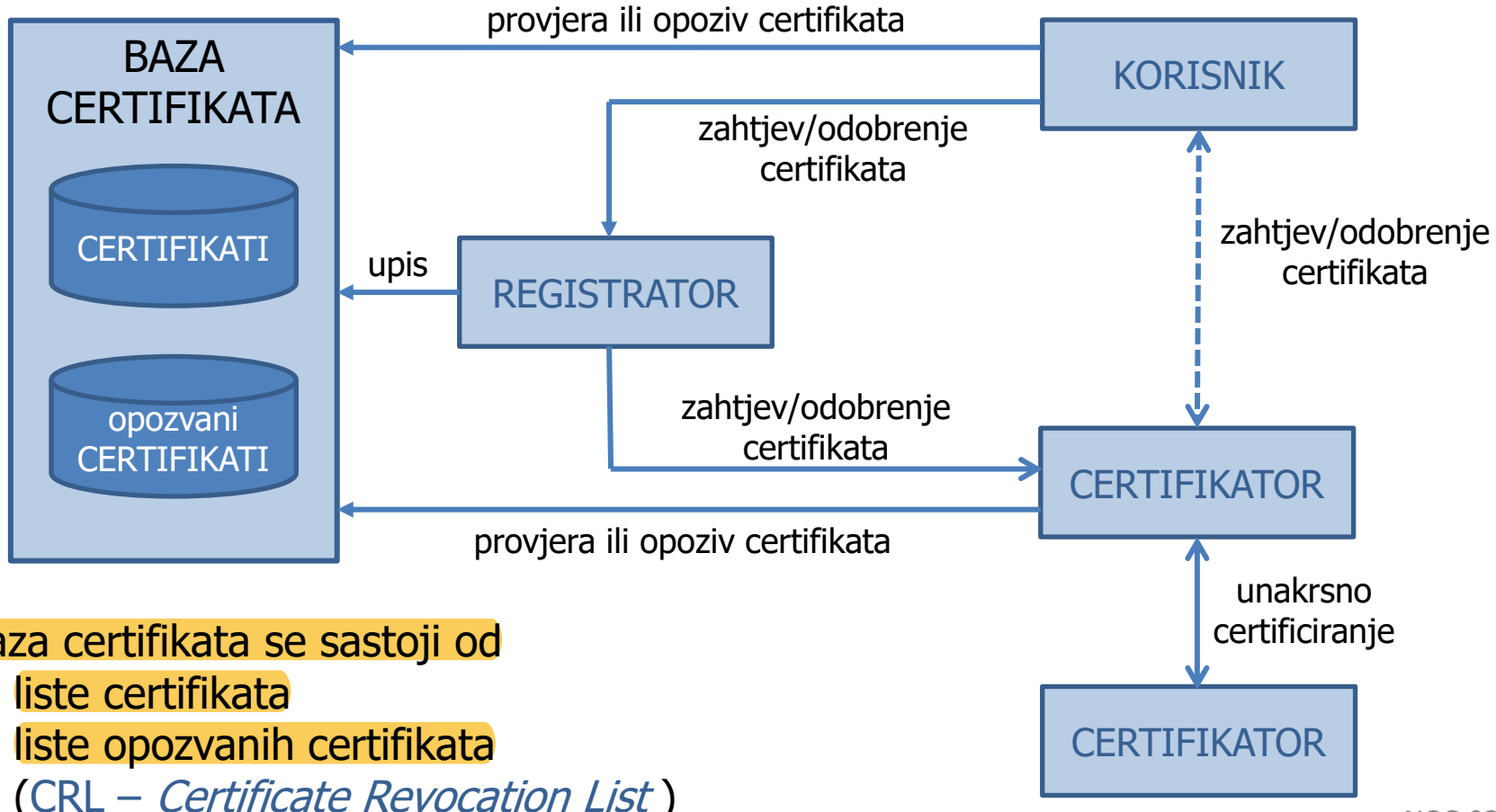
Osnovna zadaća PKI sustava

- nedvojbeno povezivanje javnih ključeva sa korisnicima te provjera jesu li ključevi trenutno važeći
- *off-line* provjera identiteta uz pomoć certifikata

Dijelovi PKI sustava

1. Korisnik
2. Certifikator (*CA - Certificate Authority*)
   - stvara i izdaje certifikate, potvrđuje da neki javni ključ pripada određenoj osobi
3. Registrator (*RA - Registration Autrhority*)
   - prima zahtjeve od korisnika, provjerava njihov identitet i prosljeđuje zahtjev CA, ali ne izdaje certifikate
   - opcionalni element PKI sustava
4. Baza certifikata
   - važeći certifikati sa datumom isteka
   - opozvani certifikati s datumom opoziva
5. Sustav za upravljanje certifikatima
   - objavljivanje, provjera, dohvat certifikata po zadanim uvjetima
6. Sustav za rekonstrukciju izgubljenih ključeva
7. Sustav za pouzdano vremensko označavanje dokumenata i potpisa (*TSA – Time Stamp Autrhority*)
   - dodaje se vremenska oznaka (time stamp)

PKI



Baza certifikata se sastoji od

- liste certifikata
- liste opozvanih certifikata
  (CRL – *Certificate Revocation List*)

Digitalni certifikat

- dokaz tko (ili što) je vlasnik javnog ključa
- skup bitnih informacija koje identificiraju korisnika (pošiljatelja) i poslužitelja (davatelja usluge)
- izdaje pouzdano certifikacijsko tijelo: izdavači cerifikata (*CA - Certificate Authorities*)
- Certifikat je svjedodžba koja potvrđuje da je određeni korisnik u trenutku izdavanja certifikata posjedovao privatni ključ koji odgovara javnom ključu u certifikatu.
- X.509
  - općeprihvaćena kriptografska norma koja propisuje sadržaj certifikata
  - koriste je mnogi Internet protokoli poput TLS/SSL

Minimalni sadržaj digitalnog certifikata

- identifikator certifikata
- osnovne podatke o nositelju certifikata
- informacije o korištenim kriptografskim algoritmima
- vrijeme i datum izdavanja certifikata
- rok valjanosti certifikata
- klasu certifikata
- identitet izdavatelja certifikata
- digitalni potpis izdavatelja certifikata i identifikaciju algoritma
- javni ključ nositelja certifikata i identifikaciju algoritma
- namjena javnog ključa nositelja certifikata

Digitalni certifikat

FER-ov digitalni certifikat od 2008 do 2018

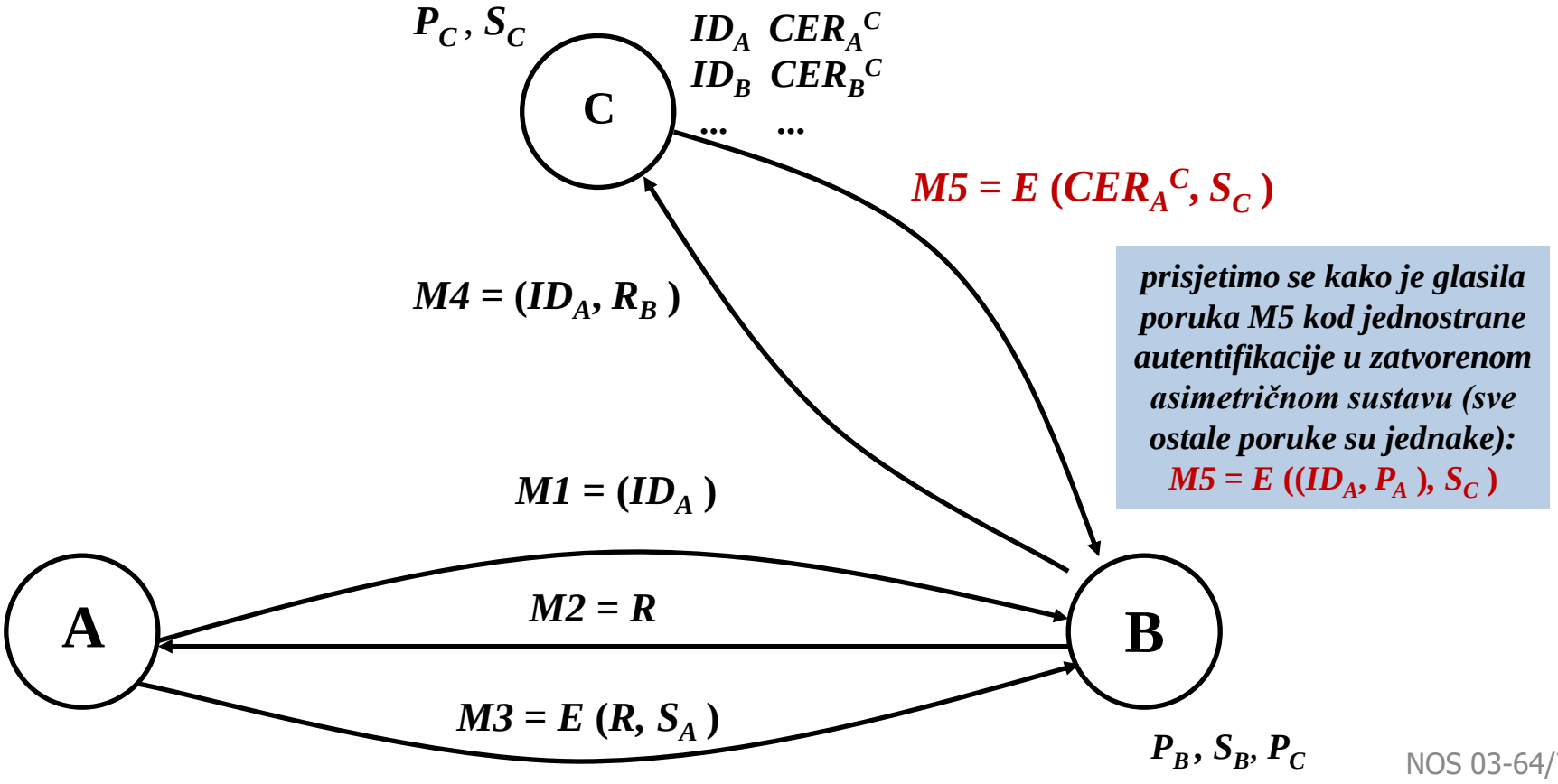
```
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number: 75 (0x4b)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=US, ST=WA, L=Seattle, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=certs@thawte.com
       Validity
           Not Before: May 13 23:33:08 2008 GMT
           Not After : Dec 31 23:59:59 2020 GMT
       Subject: C=HR, ST=Grad Zagreb, L=Zagreb, O=FER, OU=CIP,
                CN=webmail.fer.hr/emailAddress=korisnik@webmail.fer.hr
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus:
                   00:cd:66:28:fb:b8:b3:b7:e0:72:77:48:2d:08:04:
                   e1:6d:1c:c5:4f:57:73:0c:e6:db:3b:8e:cd:c6:25:
                   61:7f:60:c9:da:a3:9f:1d:fa:d8:ef:00:7b:f9:54:
                   65:ab:7e:9e:9b:6d:ff:d4:12:ad:f8:ac:87:6e:83:
                   ec:65:5f:b4:2d:eb:b8:dc:1c:d7:32:b7:46:a5:e3:
                   a1:6c:0b:4c:1b:0c:89:0a:fb:0e:3a:c0:0f:af:b2:
                   62:1d:2f:60:e4:b1:27:b4:7c:59:00:2c:19:e9:f3:
                   a3:88:fe:01:d6:56:be:26:c7:f8:42:b1:79:39:98:
                   a1:b4:4a:84:dd:20:ca:e7:a9:db:6d:a6:73:88:e7:
                   81:8b:3e:81:3d:00:e5:5d:7f:3d:9b:cd:ba:9b:28:
                   88:88:7f:d7:69:2c:66:eb:8f:79:b8:ec:bc:bb:76:
                   67:b1:00:2a:70:bd:f1:21:66:6f:ba:74:81:82:30:
                   02:c0:a8:57:f8:9f:76:02:df:7f:49:44:4a:32:93:
                   48:a4:25:73:47:10:21:20:fe:b6:d2:09:1a:60:4f:
                   a5:d9:df:ea:55:49:43:c6:ce:96:0b:7d:a7:22:c1:
                   3e:5b:28:2e:2c:04:7a:b2:93:89:db:d8:2b:59:86:
                   a3:0a:c1:6f:f9:56:b2:a5:71:4c:4b:74:f3:b8:a1:
                   b4:65
               Exponent: 65537 (0x10001)
       X509v3 extensions:
           X509v3 Basic Constraints: critical
               CA:TRUE
   Signature Algorithm: md5WithRSAEncryption
       07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
       a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
       e7:20:1b:8b:ca:a4:ab:8d:e9:51:d9:e2:4c:2c:59:a9:da:b9:
       b2:75:1b:f6:42:f2:ef:c7:f2:18:f9:89:bc:a3:ff:8a:23:2e:
       70:47
```

Digitalni certifikat

Pristup web stranicama FER-a:
The connection to this site is encrypted and authenticated using TLS 1.2, ECDHE_RSA with P-256, and AES_128_GCM.

```
Certificate:
    Data: Version: 3 (0x2)
        Serial Number: 0a:2f:ab:75:d4:a1:ee:f5:ea:df:74:15:aa:fd:47:c4
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=NL, ST=Noord-Holland, L=Amsterdam, O=TERENA, CN=TERENA SSL CA 3
    Validity  Not Before: May 13 00:00:00 2018 GMT
              Not After : May 20 12:00:00 2020 GMT
    Subject: C=HR, L=Zagreb, O=Sveu\xC4\x8Dili\xC5\xA1te u Zagrebu, OU=CIP,
      CN=*.fer.unizg.hr
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c6:bb:ca:00:b5:40:96:b3:6b:2e:94:7e:43:77:
                    39:06:d2:4f:11:c0:c4:17:e5:eb:d6:10:a5:2c:fa:
                    4c:f1:50:35:59:59:2b:fa:b5:22:26:3f:0a:ff:f2:
                    f9:c4:d7:e2:67:5d:bf:b5:c1:cc:6b:77:31:e9:de:
                    95:b0:76:53:47:f7:1f:fe:c4:5b:c1:a7:fd:c4:fc:
                    61:d3:ea:b5:28:48:e5:d5:96:a0:11:ed:0b:00:a2:
                    42:c9:fa:94:26:89:f5:37:db:0a:9a:f8:95:e8:a6:
                    35:8a:68:33:90:c2:22:10:ad:65:3a:95:5f:64:1f:
                    6f:43:88:b2:1c:f8:29:9e:51:6b:e4:2d:8c:3e:39:
                    90:f7:31:8e:32:f8:0f:cf:3e:b4:7a:c6:f3:27:17:
                    a3:4e:3c:7c:27:07:3d:68:fc:5e:9c:87:86:74:ea:
                    22:32:d5:aa:93:e4:d4:78:23:d2:88:0f:e3:8f:05:
                    8c:54:b8:95:29:eb:c2:0a:fc:26:20:ca:52:ff:ce:
                    75:6b:29:82:d6:67:06:0b:49:53:37:0d:7e:cf:1c:
                    7e:88:90:8d:7a:e7:99:fc:9f:d7:5c:e2:1f:73:19:
                    cc:27:ba:31:6f:82:40:b0:cb:8a:d2:95:f4:6e:72:
                    78:b6:02:f5:f4:0b:b6:60:32:fb:3f:34:66:f2:a4:
                    12:c5
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                keyid:67:FD:88:20:14:27:98:C7:09:D2:25:19:BB:E9:51:11:63:75:50:62
                  URI:http://crl3.digicert.com/TERENASSLCA3.crl
...
    Signature Algorithm: sha256WithRSAEncryption
         a3:aa:9b:c3:04:c3:5c:64:32:9c:8f:08:31:89:15:8a:52:19:
         fb:02:e9:dd:ab:59:3e:9e:d8:b8:52:b2:8d:df:5a:29:dc:2b:
         c0:01:7d:96:87:5c:a7:01:7e:26:c9:3b:be:01:d3:9c:71:62:
         e3:e5:a2:ce:5d:ee:59:b5:ed:20:d8:80:27:ac:af:f5:6a:73:
         79:35:d2:c5
```

Postupak jednostrane autentifikacije uz pomoć certifikata

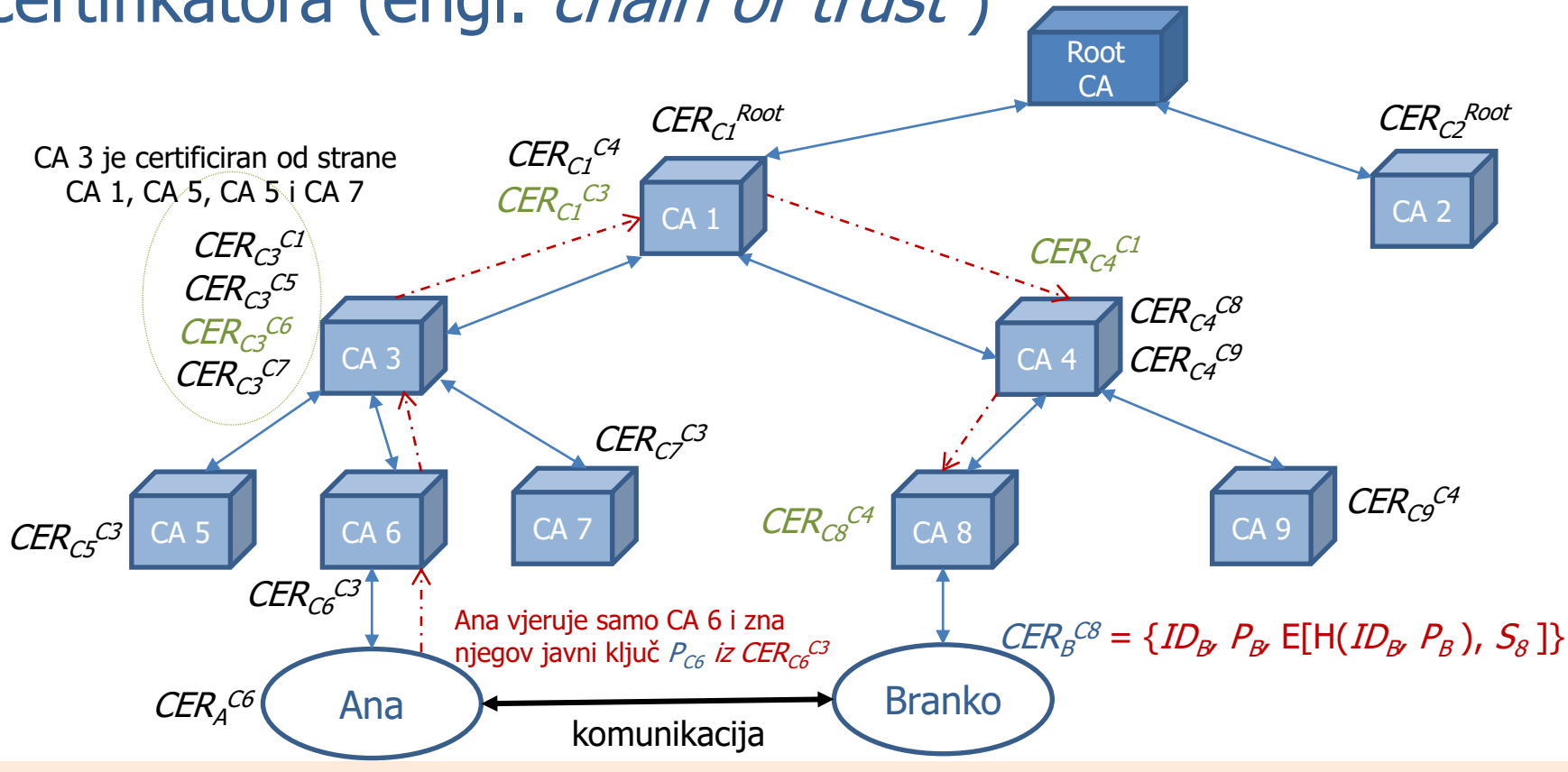


prisjetimo se kako je glasila poruka M5 kod jednostrane autentifikacije u zatvorenom asimetričnom sustavu (sve ostale poruke su jednake):
$M5 = E\ ((ID_A, P_A), S_C)$

Vrste certifikata

- prema onome čemu služe
  - potpisivanje (dokumenata, programskog koda)
  - kriptiranje
  - potpisivanje i kriptiranje
- prema načinu identifikacije prilikom inicijalizacije
  - e-mail, korisnički ID ili korisničko ime (klasa 1, engl. *class 1 digital certificate*)
    - nema nikakvu pravnu potporu jer se provjerava identitet preko e-maila
  - identifikacija subjekta preko treće strane kojoj vjerujemo (klasa 2)
  - identifikacijski dokument uz obaveznu fizičku nazočnost u CA (klasa 3)
- prema vrsti korisnika
  - obični korisnici, poslovni subjekti
  - organizacije
  - klijentske aplikacije, poslužiteljski certifikati, servisi

Provjera certifikata u hijerarhijskoj strukturi certifikatora (engl. *chain of trust*)



$$P_B = P_{C6} \bullet CER_{C3}^{C6} \bullet CER_{C1}^{C3} \bullet CER_{C4}^{C1} \bullet CER_{C8}^{C4} \bullet CER_B^{C8} = P_{C6} \bullet (A \rightarrow B) \bullet CER_B^{C8}$$

provjera certifikata CA 3

provjera redom certifikata CA 1, CA 4, CA 8 i Brankovog certifikata CER_B^{C8}

Problem opoziva certifikata

- u slučaju gubitka ili kompromitiranosti privatnog ključa, korisnik je dužan od certifikatora tražiti opoziv certifikata
- ovaj postupak je najslabija točka PKI sustava jer nije moguće istodobno obavijestiti sve zainteresirane strane
- ovaj problem nije moguće riješiti bez *on-line* veze i centralizirane baze podataka, što je u suprotnosti s idejom PKI sustava sa certifikatima
- rješenje: osigurati *on-line* vezu sa certifikacijskim centrom

Preporučeni X.509 autentifikacijski protokoli

- protokol s jednom porukom (*one - way protocol*)
  - autentificiraju se oba sudionika *A* i *B*
  - osigurava integritet sadržaja koji se prenosi sudioniku *B*
  - uporabom vremenske oznake sprječava napad ponavljanjem poruke
- protokol s dvije poruke (*two - way protocol*)
  - pridodaje se odgovor sudionika *B*
  - utvrđuje da je upravo sudionik *B* a ne neki napadač odgovorio na prvu poruku
  - uporabom vremenske oznake sprječava napad ponavljanjem druge poruke
- protokol s tri poruke (*three - way protocol*)
  - sudionik *A* vraća treću poruku sudioniku *B*
  - ne upotrebljavaju se u svim porukama vremenske oznake

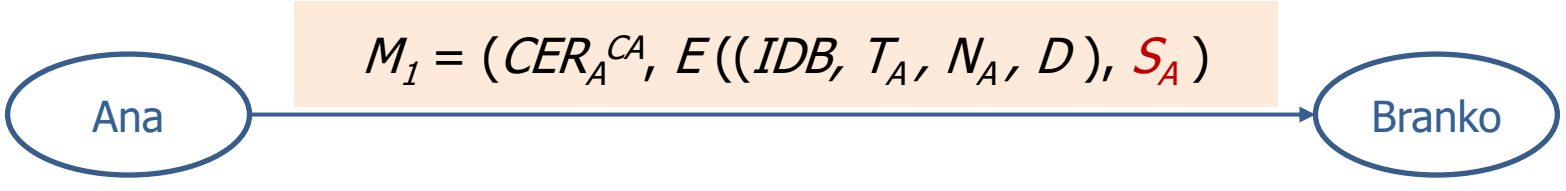
Protokol s jednom porukom (*one - way protocol*)

1. Kada *Ana (A)* želi komunicirati s *Brankom (B)* :

- generira nasumični broj N_A i oblikuje vremensku oznaku T_A (vrijeme, trajanje valjanosti oznake)
- pronalazi put $A \rightarrow B$ te iz certifikata CER_B^{CB} saznaje P_B

$$P_B = P_{CA} \bullet (A \rightarrow B) \bullet CER_B^{CB}$$

- oblikuje četvorku (IDB, T_A, N_A, D), gdje je D podatkovna komponenta koja može biti kriptirana sa P_B
- šalje *Branku* poruku:



Protokol s jednom porukom (*one - way protocol*)

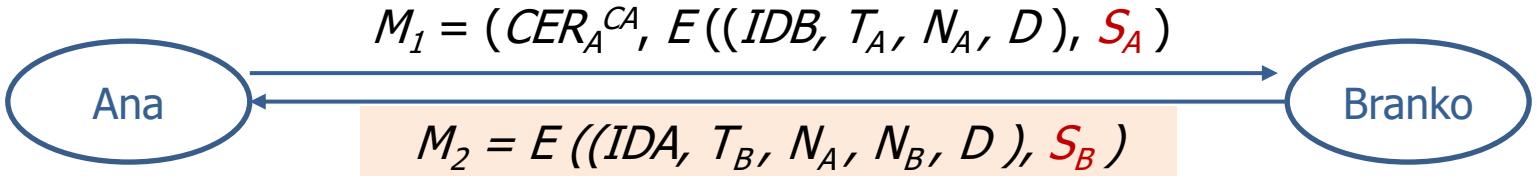
2. Kada *Branko* primi poruku M_1:
   - pronalazi u tablicama put $B \rightarrow A$ te iz CER_A^{CA} saznaje i utvrđuje *Anin* javni ključ:

$$P_A = P_{CB} \bullet (A \rightarrow B) \bullet CER_A^{CA}$$

   - uz pomoću ključa P_A dobiva (IDB, T_A, N_A, D)
   - na temelju IDB utvrđuje da je poruka stvarno upućena njemu
   - na temelju vremenske oznake T_A utvrđuje da je poruka još valjana
   - dekriptira svojim privatnim ključem S_B podatkovnu komponentu D ako je bila kriptirana
   - može usporediti dobiveni N_A s pohranjenim nasumičnim brojevima iz prethodnih poruka kako bi ustanovio da poruka nije ponovljena

Protokol s dvije poruke (*two - way protocol*)

3. *Branko* :
   - generira N_B i vremensku oznaku T_B ;
   - oblikuje (IDA, T_B, N_A, N_B, D), gdje D može biti kriptiran s P_A
   - šalje *Ani* poruku M_2 :

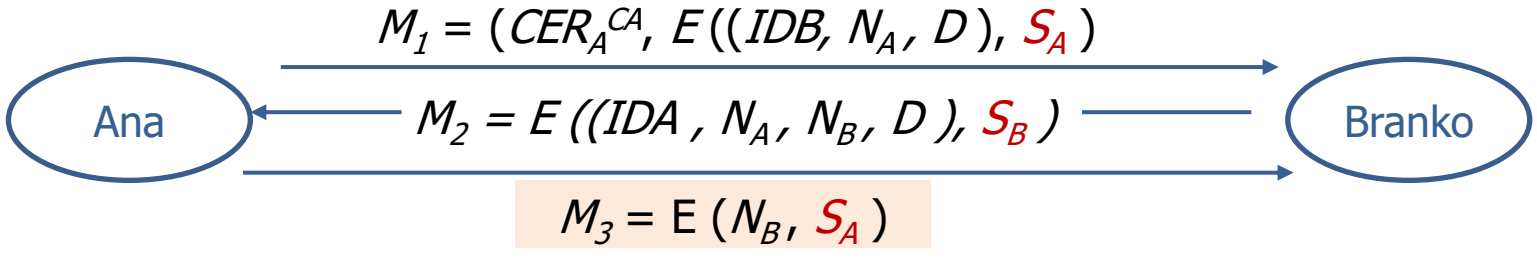


4. Kada *Ana* primi poruku M_2 :
   - uz pomoću P_B dobiva (IDA, T_B, N_A, N_B, D)
   - na temelju IDA utvrđuje da je poruka upućena baš njoj
   - na temelju T_B utvrđuje da je poruka još valjana
   - po potrebi dekriptira D uz pomoć S_A
   - može usporediti N_B s pohranjenim brojevima iz prethodnih poruka kako bi ustanovio je li poruka ponovljena

Protokol s tri poruke (*three - way protocol*)

- u prethodnim porukama M_1 i M_2 ignorira vremenske oznake T_A i T_B

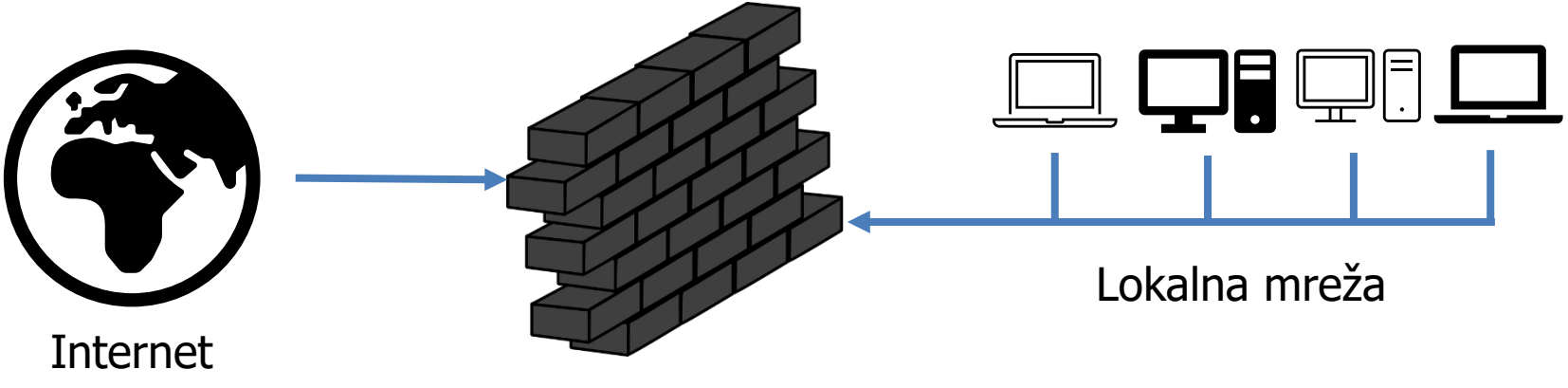
6. *Ana* :
   - uspoređuje dobiveni N_A iz poruke M_2 s izvornom vrijednošću i utvrđuje da je poruka M_2 odgovor na M_1
   - uz pomoću ključa S_A kriptira dobiveni NB i šalje poruku $M_3 = E\,(N_B, S_A)$



6. Po primitku poruke M_3 *Branko* :
   - uz pomoću ključa P_A dekriptira poruku M_3 i dobiva N_b te uspoređuje dobiveni N_B iz poruke M_3 s izvornom vrijednošću i utvrđuje da je M_3 odgovor na M_2

3.6. Sigurnosna stijena

- vatrozid, engl. *firewall*
- računalo ili neka nakupina komunikacijskih naprava koje fizički razdvajaju dvije mreže
- uobičajeno, sigurnosni štit ograničava pristup nekoj privatnoj lokalnoj mreži (ili čak samo jednom računalu) iz javne mreže



Načini djelovanja sigurnosne stijene

Osnovna funkcionalnost

- filtriranje paketa
  - statičko (bez stanja, engl. *stateless inspection*)
    - prema svojstvima
    - paketi se filtriraju nezavisno
  - dinamičko (sa stanjem, engl. *statefull inspection*)
    - vodi se evidencija kojoj vezi koji paket pripada

Dodatne mogućnosti u kombinaciji s nekim drugim uređajem

- pretvorba mrežnih adresa (engl. *Network Address Translation, NAT*)
- virtualne privatne mreže (engl. *Virtual Private Networks, VPN*)
- sustavi za otkrivanje napada (engl. *Intrusion Detection Systems*)
  - filtriranje prema sadržaju paketa - paketi se mogu pregledati sadrže li zloćudan kod
- posrednički (engl. *proxy firewall*)
- mada sigurnosna stijena i NAT/VPN/IDS/proxy nisu isti uređaji, često se oni kombiniraju

Statičko filtriranje paketa

- vrsta protokola
  - *User Datagram Protocol (UDP),*
  - *Transmission Control Protocol (TCP),*
  - *Internet Control Message Protocol (ICMP),*
  - *Internet Group Management Protocol (IGMP), ...*
- IP adrese odredišta ili izvorišta
- pristup (engl. *port*)
  - prema odredišnim i izvorišnim pristupima
  - ukupni broj pristupa je 65536
  - prvih 1024 pristupa su rezervirana za određene aplikacije i ne mogu se koristiti u neke druge svrhe, npr.
    - HTTP koristi pristup 80,
    - FTP koristi pristup 20 i 21,
    - DNS koristi pristup 53 itd...
    - neki protokoli su izrazito osjetljivi na mrežne napade pa se mogu onemogućiti poput pristupa aplikacije Telnet
- ruta usmjeravanja paketa (engl. *source routing*)